TABLE OF CONTENTS

NASA Contractor Report 3015

Development and Flight Tests of a Kalman Filter for Navigation During Terminal Area and Landing Operations

Stanley F. Schmidt, Paul F. Flanagan,
and John A. Sorenson

CONTRACT NAS2-8862
JULY 1978

NASA

NASA Contractor Report 3015

Development and Flight Tests of a Kalman Filter for Navigation During Terminal Area and Landing Operations

Stanley F. Schmidt, Paul F. Flanagan,
and John A. Sorenson
Analytical Mechanics Associates, Inc.
Mountain View, California

Prepared for
Ames Research Center
under Contract NAS2-8852

National Aeronautics
and Space Administration

Scientific and Technical
Information Office

1978

BLANK PAGE

DEVELOPMENT AND FLIGHT TESTS OF A KALMAN FILTER FOR NAVIGATION DURING TERMINAL AREA AND LANDING OPERATIONS

Stanley F. Schmidt, Paul F. Flanagan,
and John A. Sorensen

Analytical Mechanics Associates, Inc.
Mountain View, California 94043

SUMMARY

A Kalman filter for aircraft terminal area and landing navigation was implemented and flight tested in the NASA Ames STOLAND avionics computer on board a Twin Otter aircraft. This system combines navaid measurements from TACAN, MODILS, air data, baro-altimeter, and radar altimeter sensors with strap-down accelerometer measurements and attitude angles obtained from the attitude and heading reference gyros.

The flight test consisted of five approach, landing, and climbout profiles. The aircraft position and velocity were estimated simultaneously by both the Kalman filter and the regular STOLAND complementary filter. The errors in position and velocity, as determined by tracking radar, were used to evaluate and compare the two filters' performances. Post flight simulation studies were also made to identify an improved configuration of the Kalman filter.

It was shown that it is feasible to use a Kalman filter during the landing phase of flight for navigation computations. It was found that the Kalman filter improves the accuracy of the state variable estimates to some extent. The test pilot reported that on one of the flight profiles, one of the best localizer tracking performances that he had ever observed was obtained with the Kalman filter engaged. However, the computer implementation requirements of the Kalman filter are somewhat larger than those of the standard complementary filter. The results of this study form the basis for current and future navigation studies conducted at NASA Ames Research Center.

BLANK PAGE

BLANK PAGE

LIST OF FIGURES

LIST OF FIGURES (Continued)

I

INTRODUCTION

Objectives

This report describes the design and flight test results of a Kalman filter used for aircraft terminal area and landing navigation. The research was conducted using the NASA Ames STOLAND avionics system as a test bed for the Kalman filter implementation. The navigation aids used in the system include TACAN for the terminal area phase and MODILS (a low-cost experimental version of the Microwave Landing System (MLS)) for the final approach and landing phase. Other navaid information used from on board sources include air data, barometric altitude and radar altitude. The Kalman filter combines the navaid information with that from a low accuracy inertial reference system comprised of a conventional attitude and heading reference (vertical and directional gyros) and a three-axis strapdown (body mounted) accelerometer package.

One of the prime objectives of the study effort was to demonstrate that the Kalman filter used for terminal area navigation could be mechanized in a manner such that the computer requirements are not prohibitive. This was demonstrated by having a Kalman filter operating in parallel in the same airborne computer where the STOLAND system's normal functions of navigation, guidance, control, and display were also being performed. As is seen later in this report, the Kalman filter outputs can be used to drive display and guidance logic with sufficient accuracy for automatic landing of the aircraft.

In the landing sequence of the aircraft, the measurement processing rate needs to be moderately high in order to provide the required navigation accuracy. Algorithms referred to as complementary filters have been employed [1,2] in the STOLAND system to combine dead-reckoning and position measurement data. These algorithms require only a modest number of computer operations. Hence, the complementary filters can typically process the position measurements at the high frequency rate. Because the complementary filters exist as a part of the normal STOLAND navigation system, the navigation performance, computation cycle, and memory requirements of the Kalman and complementary filters could directly be compared.

The Kalman filter was previously used in the experimental RAINPAL system [3,4] which demonstrated that the high navigation accuracy required for automatic landing is achievable when using an accurate inertial (platform-based) system and precision ranging navaids. The system described herein used a number of Kalman filter mechanization details developed for the RAINPAL system such as:

(a) the square root implementation,

(b) computer time-sharing logic for maximizing available machine time for the filter, and

(c) approximations and logical implementations which save computer operations.

The distinction of the navigation system using the Kalman filter described in this report is the lower accuracy of the inertial (strapdown-based) system and the navaids used by the filter.

Background

This research originally began [5] with the design of a simple Kalman filter to operate in the horizontal (x - y) plane for potential use in a STOL aircraft landing navigation system. Computer simulation results (based on artificially generated approach trajectories and instrument errors) showed that the Kalman filter produced much smaller navigation errors than did a complementary filter using TACAN data. The same result occurred during transition from TACAN to scanning beam ILS (MODILS) data. The Kalman filter was shown to gain its superior performance at the expense of real time computation and memory required. However, these additional computer requirements were not considered excessive.

Because of these encouraging results, the Kalman filter was expanded to include the vertical axis [6]. The resulting three-axis Kalman filter was designed to be interfaced with the displays and controls of STOLAND. The filter design was then validated on the NASA Ames CDC 7600 computer using recorded flight data. The error characteristics of both the Kalman filter and STOLAND complementary filter were compared and discussed. In addition, a comprehensive study effort has been undertaken by NASA Ames scientists on the application of Kalman filtering to terminal area navigation. This more general study effort confirms that the particular design selected for this effort is appropriate for the STOLAND system. Documentation of this NASA study will appear at a future date.

As a result of the success of these three efforts, it was decided to mechanize the three-axis Kalman filter in the STOLAND system, and then to flight test it on the NASA Ames Twin Otter aircraft. This study was designed to examine in detail the performance and implementation requirements of using a Kalman filter for terminal area navigation. This report summarizes the filter designs, documents the results of the flight tests, and presents further post-flight analyses.

Report Overview

Chapter II of this report presents the notation used throughout the report, and it defines various symbols, acronyms, and abbreviations.

In Chapter III, the on board test system used in the study is described. This includes discussion of the STOLAND avionics system and its associated complementary filters. Also included are a brief general description of the three-axis Kalman filter used in the study and a comparison of the implementation requirements for the Kalman and complementary filters.

In Chapter IV, the flight test results are presented. These include descriptions of the test facilities, data processing procedures, flight data collected, test results, and pilot comments.

In Chapter V, post flight studies to explain test anomalies and to improve filter performance are explained.

Chapter VI presents some concluding remarks.

Appendix A gives the mathematical details of the Kalman filter used in this study, and Appendix B presents some of the on board mechanization details.

This work was carried out for NASA Ames Research Center under the technical management of Mr. Rodney Wingrove. The STOLAND flight system used in the tests was under the direction of Mr. Don Smith. The NASA test pilot responsible for the overall evaluation on the STOLAND system simulator and for the flight tests was Mr. Gordon Hardy. The authors wish to express appreciation to the above individuals for their contributions in the overall development and flight tests of the Kalman filter system described herein.

BLANK PAGE

II

NOTATION AND DEFINITIONS

Notation

The notation of "·" over a symbol has the customary meaning of differentiation with respect to time. The "^" (hat) mark over a symbol means the "estimated" or "computed" value of the symbolized quantity. The letter "d" before a quantity indicates an error or small variation of that quantity. For example, if X is the true value of position, it may be written as the sum of the estimated position and the position error, or

$$X = \hat{X} + dx \quad .$$

The notation t_k, t_{k+1}, etc., are used to denote discrete points in time. The time point t_{k+1} occurs Δt seconds after t_k, or

$$t_{k+1} = t_k + \Delta t \quad .$$

The time differential Δt denotes the primary cycle time of the implemented digital filter.

Other notations include:

$(\ )^T$ = transpose of matrix;

$E\{ \ \}$ = expected value of enclosed quantities;

$(\ )_x, (\ )_z$ = matrix referred to the x-y portion of the Kalman filter and matrix referred to the decoupled z portion of the filter;

$(\ )_a, (\ )_b$ = computed quantity after (a) and before (b) an update is added;

$(\ )_{res}$ = residual, or difference between what is considered the correct value of a quantity (e.g., determined by tracking radar) and the value determined from the filter.

Roman Symbols

A	-	discrete form of the matrix F_x.
a_{sx}, a_{sy}, a_{sz}	-	sum of raw acceleration measurements and estimated acceleration biases.
$A_x, A_y, A_z;$ a_{xb}, a_{yb}, a_{zb}	-	aircraft acceleration measured in body axes.
B, C, D	-	temporary matrices used to update the square root covariance W.
b_{ax}, b_{ay}, b_{az}	-	acceleration measurement biases.
$\hat{b}_{ax}, \hat{b}_{ay}, \hat{b}_{az}$	-	estimates of acceleration measurement biases.
$b_h, \hat{b}_h$	-	actual and estimated bias errors in the barometric altitude measurement.
$b_r, \hat{b}_r$	-	actual and estimated bias error in the TACAN range measurement.
$b_\psi, \hat{b}_\psi$	-	actual and estimated bias error in the TACAN bearing measurement.
c	-	constant used to test reasonableness of a measurement residual.
c_x, c_v	-	position and velocity smoothing vectors.
dx	-	the n element continuous error state vector of the estimate $\hat{X}$.
$d\hat{x}$	-	filter estimate of the error state vector dx.
F_x	-	$n \times n$ system dynamics matrix.
F_η	-	$n \times m$ error distribution matrix.
$H. H_m$	-	external measurement distribution (sensitivity) matrix.
H_{ax}, H_{ay}	-	sensitivity vectors of true airspeed components to estimated state.
H_e	-	sensitivity vector of altitude derived from the MODILS elevation measurement to estimated state.

H_h - sensitivity vector of baro-altitude to estimated state.

H_{ma} - sensitivity vector of MODILS azimuth to estimated state.

H_{mr} - sensitivity vector of MODILS range to estimated state.

H_{ms} - accumulated measurement sensitivity matrix.

H_r - sensitivity vector of radio altimeter measurement to estimated state.

H_{tb} - sensitivity vector of TACAN bearing to estimated state.

H_{tr} - sensitivity vector of TACAN range to estimated state.

h_B, h_R - baro-altitude and radio altitude.

h_r - runway altitude with respect to sea level.

h_t - altitude above the TACAN station.

I - identity matrix.

K_m - Kalman filter gain matrix.

k_{1x}-k_{4x} - complementary filter gains for the x direction.

k_{1z}-k_{4z} - Kalman and complementary filter gains for the z direction.

N_M - magnetic north.

n_x - number of residuals in a sum.

P - covariance matrix of the error state dx.

Q - assumed variance of the random error q.

Q_{ax}, Q_{ay} - assumed variances of air data measurement noise.

Q_e - assumed variance of MODILS (elevation derived) altitude measurement.

Q_h - assumed variance of baro-altitude noise.

Q_{ma} - assumed variance of MODILS azimuth measurement noise.

Q_{mr} - assumed variance of MODILS range measurement noise.

Q_r - assumed variance of radio altimeter noise.

Q_s - assumed variance of random error in the accumulated residual.

Q_{tb} - assumed variance of TACAN bearing measurement noise.

Q_{tr} - assumed variance of TACAN range measurement noise.

Q_{xx} - variance of an individual measurement.

q - random noise error in the external measurement.

q_{ax}, q_{ay} - random noise in the air data measurements.

q_e - random noise in the MODILS (elevation derived) altitude measurement.

q_h - random noise error in the baro-altitude measurement.

q_{ma} - random noise error in MODILS azimuth measurement.

q_{mr} - random noise in MODILS range measurement.

q_r - random noise in the radio altimeter measurements.

q_{tb} - random noise error in the TACAN bearing measurement.

q_{tr} - random noise error in the TACAN range measurement.

R_s - TACAN or MODILS measured slant range.

r, r_1 - MODILS slant range and its component in the x-z plane.

r_c - TACAN measured ground range.

S - variance of the accumulated residual of an individual measurement.

t - time.

U - standard deviation of u.

u - a constant vector over the time interval Δt for approximating the effect of the random vector η.

V_a, V_T - true airspeed.

$\hat{v}, \hat{v}_s$ - smoothed velocity vector.

$\hat{v}_r$ - estimated velocity vector.

v_x, v_y, v_z - ground velocity components along, normal, and vertical to runway.

$\hat{v}_x, \hat{v}_y, \hat{v}_z$ - estimates of v_x, v_y, v_z.

W - the square root of P.

w_x, w_y - wind velocity components along and normal to runway.

$\hat{w}_x, \hat{w}_y$ - estimates of w_x, w_y.

$X, \hat{X}$ - actual and estimated values of the aircraft's state vector.

x,y,z - position of aircraft in a Cartesian reference frame with x along the runway, y in the horizontal plane normal to the runway, and z normal to the horizontal plane and positive pointing downward.

$\hat{x}, \hat{y}, \hat{z}$ - filter estimates of x,y,z.

x_E, y_E, z_E - location components of the MODILS elevation antenna with respect to the runway reference frame.

x_e, y_e, z_e - aircraft position coordinates with respect to the MODILS elevation antenna.

$\dot{\hat{x}}_a, \dot{\hat{y}}_a$ - horizontal true airspeed components in the runway reference frame.

x_m, y_m, z_m - location components of the MODILS transponder and scanner with respect to the runway reference frame.

x_r, y_r, z_r - aircraft position coordinates with respect to the runway reference frame.

$\ddot{x}_r, \ddot{y}_r, \ddot{z}_r$ - raw acceleration in the runway reference frame as computed by STOLAND system software.

$\hat{x}_r$ - estimated position vector.

$\hat{x}_s$ - smoothed position vector.

x_T, y_T, z_T - location components of the TACAN ground station with respect to the runway reference frame.

$\hat{Y}$ - computed value of the external measurement.

Y_{ax}, Y_{ay}, $\hat{Y}_{ax}, \hat{Y}_{ay}$ - actual and estimated x and y components of true airspeed.

$Y_e, \hat{Y}_e$ - actual and estimated MODILS altitude measurements.

Y_m - external state measurement of aircraft.

$Y_{ma}, \hat{Y}_{ma}$ - actual and estimated MODILS azimuth measurements.

$Y_{mr}, \hat{Y}_{mr}$ - actual and estimated MODILS range measurements.

$Y_r, \hat{Y}_r$ - actual and estimated radio altimeter measurements.

$Y_{tb}, \hat{Y}_{tb}$ - actual and estimated TACAN bearing measurements.

$Y_{tr}, \hat{Y}_{tr}$ - actual and estimated TACAN range measurements.

y_m - discrete measurement residual.

$\hat{y}_m$ - computed value of the position measurement residual based on the error state estimate $\hat{dx}$.

y_s, y_{sm} - accumulated residual Δy.

Greek Symbols

α, β - constants used to compute smoothing vectors c_x and c_v.

$\hat{\gamma}$ - estimated flight path angle

Δt - major time (cycle) update period of the Kalman filter.

Δt_f - acceleration integration period of the filter.

Δv_c - commanded change in airspeed.

Δy - error in the estimated residual $\hat{y}_m$.

$\Delta\psi$ - difference between the TACAN measured bearing and the bearing of the runway.

ε, ϵ - MODILS elevation angle measurement.

η - vector of m random forcing functions for compensation of error growth caused by unmodeled error sources.

τ_a - time constant for acceleration measurement colored noise.

τ_h - time constant for barometric altimeter measurement colored noise.

τ_r - time constant for TACAN range measurement colored noise.

τ_w - time constant for wind error measurement colored noise.

τ_x, τ_v - time constants used to compute smoothing vectors c_x and c_v.

τ_ψ - time constant for TACAN bearing measurement colored noise.

σ_{ax}, σ_{ay} - standard deviation (std) of acceleration colored noise.

σ_{hb} - std of barometric altitude colored noise (bias).

σ_{hr} - std of barometric altitude random noise.

σ_{m} - std of the m^{th} residual sum y_{sm}.

σ_{qr} - std on TACAN range random noise.

$\sigma_{q\psi}$ - std in TACAN bearing random noise.

σ_{r} - std of TACAN range colored noise (bias).

σ_{v} - std of velocity noise.

σ_{va} - std of air data velocity noise.

σ_{vz} - std of vertical airspeed error.

σ_{wx}, σ_{wy} - std of wind colored noise (bias) components.

σ_{ψ} - std of TACAN bearing colored noise (bias).

$\sigma_{\psi i}$ - std in initial heading measurement.

$\Phi(t_{k+1}; t_k)$ - state transition matrix from time point t_k to time point t_{k+1}.

$\Phi_u(t_{k+1}; t_k)$ - forcing function sensitivity matrix affecting state at t_{k+1} due to $u(t_k)$.

φ, θ, ψ - aircraft attitude angles (roll, pitch, heading) measured by vertical and directional gyros.

φ_c, θ_c - commanded aircraft attitude.

ψ_M - MODILS azimuth measurement.

ψ_r, ψ_{RW} - bearing of runway with respect to magnetic north.

ψ_T - TACAN measured bearing from magnetic north.

Abbreviations and Acronyms

EADI	-	Electronic Attitude Director Indicator.
HSI	-	Horizontal Situation Indicator.
MFD	-	Multifunction Display.
MODILS	-	mobile, scanning beam instrument landing system.
RAINPAL	-	recursived aided inertial navigation for precision approach and landing.
std	-	standard deviation.
STOLAND	-	integrated avionics system for STOL aircraft.
TACAN	-	tactical aircraft navigation aid, providing range and bearing measurements.

BLANK PAGE

III

TEST SYSTEM DESCRIPTION

This chapter first presents a brief summary of the STOLAND flight test system which was used to conduct the study described in this report. This summary includes an overview of the STOLAND system avionics and a description of its complementary filters normally used for terminal area navigation. This is followed by a summary of the design considerations and a brief description of the Kalman filter which was added to the STOLAND software to provide the experimental, alternate terminal area navigation system which is the subject of this study. Then, the computer mechanization requirements for implementing the Kalman filter and the complementary filter are compared.

The STOLAND System

General description.- The NASA Ames STOLAND system is an integrated digital avionics package designed for testing terminal area guidance, navigation, control, and display concepts and for investigating operational procedures for short-haul aircraft [7-9]. The STOLAND system can be tested in either the Ames Augmentor Wing or the Twin Otter STOL aircraft. Flight tests of this system are typically conducted at the NASA Ames Crows Landing facility in central California where tracking radar is provided.

Figure 1 presents a pictoral description of the main units of the STOLAND flight test system. The subsystem interconnections are illustrated in further detail by the block diagram in Fig. 2. The general computational flow of the navigation, guidance, and flight control functions are depicted in Fig. 3 [9].

As is illustrated in Figs. 1-3, the STOLAND system has a moderately large number of navaid receivers, on board sensors, and pilot control and command inputs which are interfaced with the Sperry 1819A airborne computer. This computer is the heart of the overall mechanization, and it is used to provide both pilot assist modes and completely automatic modes for flying the aircraft. The computer is interfaced with various displays (EADI, MFD, HSI), control sensors, mode select and data entry panels, steering column, navaids (TACAN, MODILS), vehicle sensors, and the data acquisition system through a data adapter.

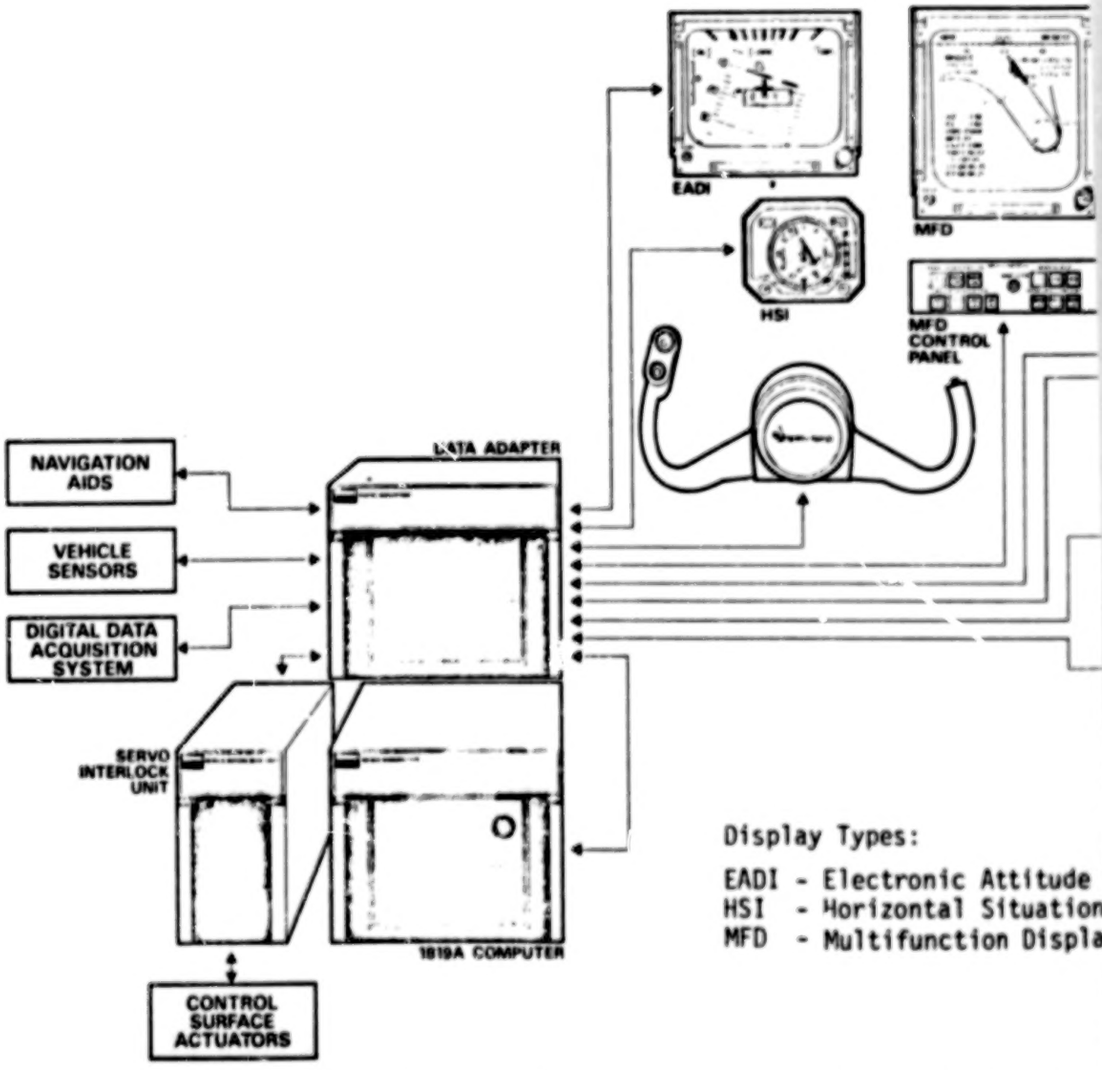
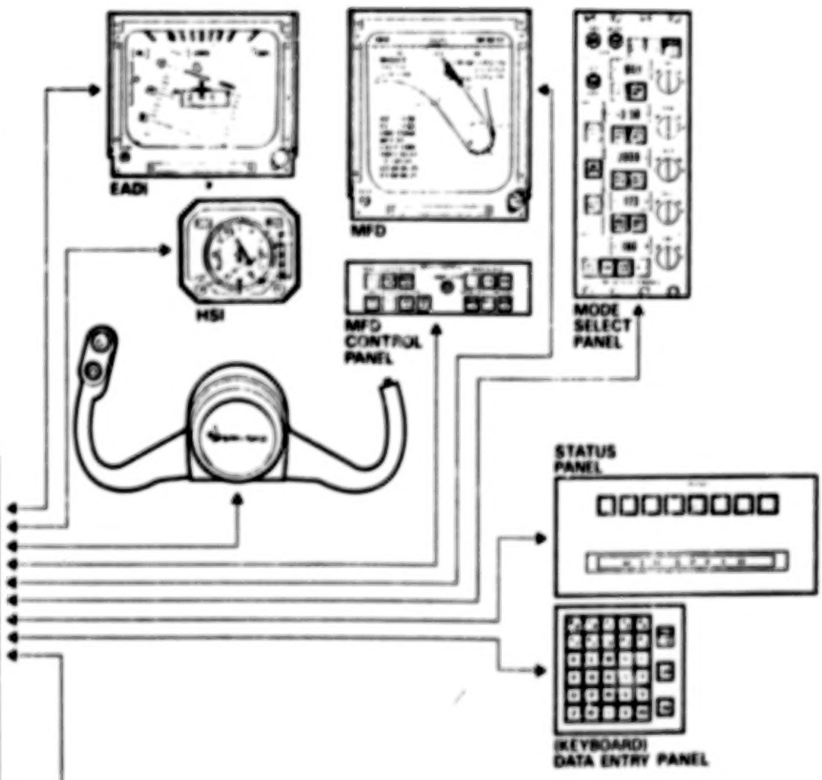


FIGURE 1.- PICTORAL DESCRIPTION OF THE STOLAND FLIGHT TEST S



Display Types:

EADI - Electronic Attitude Director Indicator
HSI - Horizontal Situation Indicator
MFD - Multifunction Display

TION OF THE STOLAND FLIGHT TEST SYSTEM

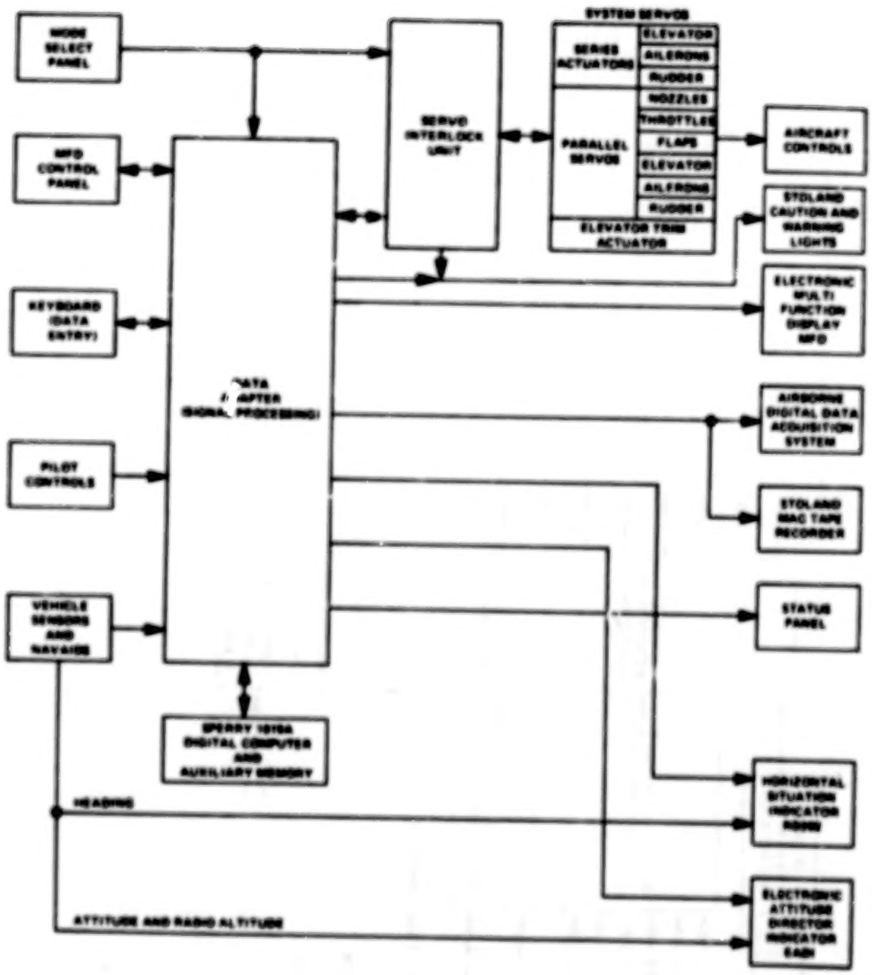


FIGURE 2.- BLOCK DIAGRAM OF THE STOLAND SYSTEM COMPONENTS [8]

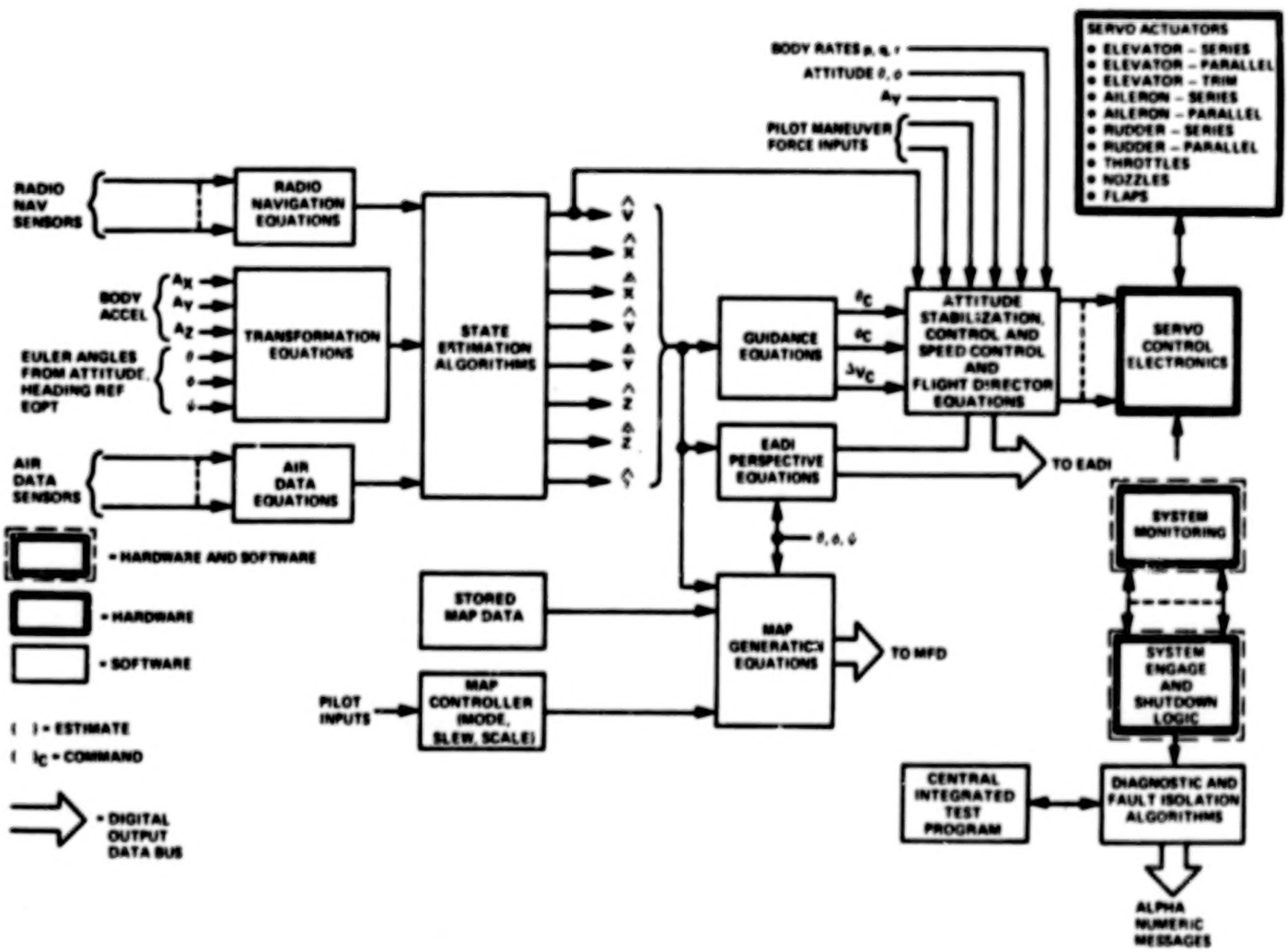


FIGURE 3.- SIMPLIFIED FLOW OF STOLAND COMPUTATION FUNCTIONS [8]

The Sperry 1819A computer used in the STOLAND system implementation contains 32,768 words of 18-bit memory. The speeds of individual operations for single and double precision fixed point arithmatic on the 1819A are as follows:

Operation	Time (μsec)
ADD (Single)	4
ADD (Double)	6
Multiply (Single)	24
Divide (Single)	24
LOAD and STORE (Single)	4
LOAD and STORE (Double)	6

The STOLAND 1819A software is divided into many functions which are presented in Table 1 [9]. Also shown in this table are the computation rate at which each of these functions is performed, the real time consumption required for each function, and the memory usage division among the functions. Note, in particular, that the STOLAND system navigation function, performed at a rate of 20 Hz, requires 0.12 sec of execution time for every one second of real time operation, and it requires 1,066 memory words.

Also, note in Table 1 that there are about 165 msec of real time available each second and 10,000 words of memory that are unused which can be used for additional avionics functions. This available time and memory were used to implement the Kalman filter based navigation software.

Complementary navigation filters.- The current STOLAND navigation system consists of Sperry designed complementary filters [2] which combine raw measurement data from external navigation aids (TACAN, MODILS), with on board sensors (air data, radar altimeter, linear accelerometer, and altitude and heading reference). The STOLAND navigation equations provide position and velocity of the aircraft with respect to a runway-fixed Cartesian coordinate system.

Figure 4 is a planar view of the Crows Landing test facility where the test flights were conducted [1]. The coordinate system has its origin at the center of the runway and the intersection of the perpendicular from the position of the scanning beam ILS receiver (MODILS) elevation antenna (glideslope) to the runway centerline. The x axis is along the runway centerline, the y axis is away from the MODILS elevation antenna, and the z axis is downward, completing the Cartesian set.

TABLE 1.- STOLAND SOFTWARE SUMMARY [8]

Function	Time/ Solution (msec)	Iteration Rate	Time Consumption (msec/sec)	Memory Usage (words)
Master Executive and Timing	1.2	20/sec	24.0	374
Input/Output	5.6	20/sec	112.0	339
Monitors and Diagnostics	1.5	20/sec	30.0	619
Keyboard and Status Panel Number Entry	.052	20/sec	1.04	1,453
Decode/Display	2.304	20/sec	46.08	
Mode Select panel and Mode Interlocks	1.3	10/sec	13.0	1,762
Navigation	6.0	20/sec	120.0	1,066
Air Data Computation (h, V_C, V_T, Q, T_T, T_S..)	1.3	20/sec	26.0	132
Attitude Stabilization Control Stick Steering and Flight Director	3.7	20/sec	74.0	880
Autopilot and Autopilot Executive (includes Trim Tables)	4.0	20/sec	80.0	3,134
Electronic ADI (including Runway Perspective Display)	4.19	20/sec	83.8	561
3-D Guidance	2.0	10/sec	20.0	1,785
4-D Guidance	1.2	.1/sec to 10/sec	12.0 (max)	1,304
Multifunction Display (MFD)	6.0	1.0/sec and 20/sec	120.0 (max)	3,636
Horizontal Situation Indicator	2.3	10/sec	23.0	230
Magnetic Tape and Digital Data Acquisition	1.2	20/sec	24.0	495
Speed Control....Autothrottle Flap, Nozzle, etc	1.4	20/sec	28.0	556
Data for all Modules (except 3D-4D and MFD)	---	---	---	3,542
Totals			835.92 absolute max	21,868
Preflight Central Integrated Test				2,558

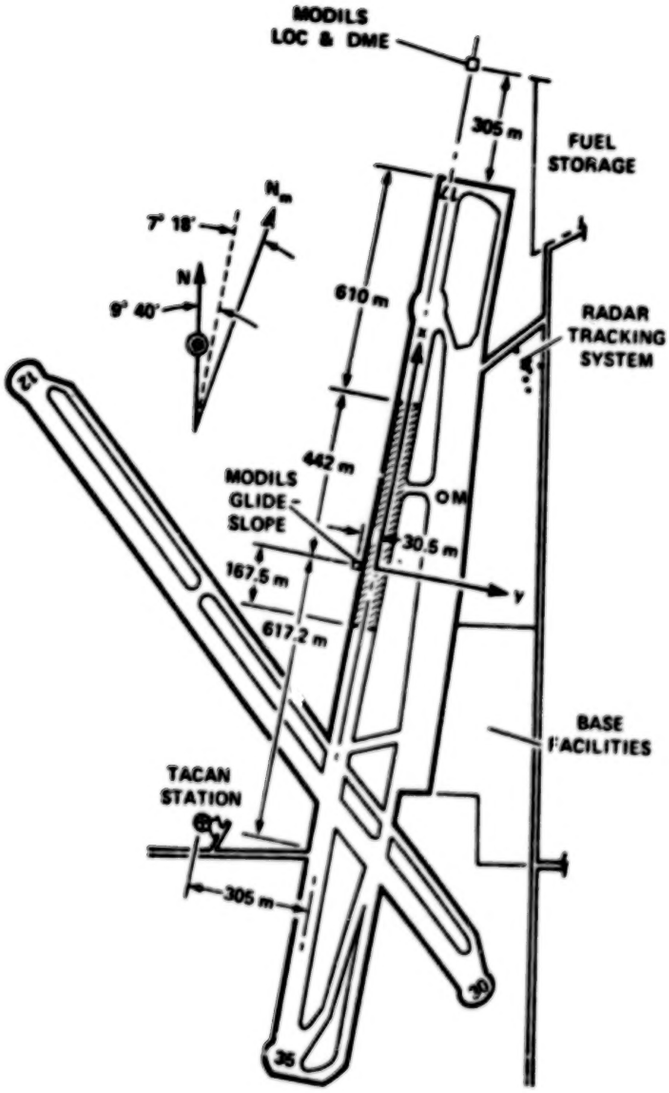


FIGURE 4. PLANE VIEW OF CROWS LANDING TEST FACILITY SHOWING THE REFERENCE COORDINATE SYSTEM AND THE NAVAID LOCATIONS [1]

Figure 5 defines the coordinates of the TACAN station which provides measurements of slant range R_s and bearing ψ_T with respect to the aircraft. The geometry of the MODILS system is depicted in Fig. 6. MODILS provides measurements of slant range R_s, azimuth ψ_M, and elevation ε. Both MODILS antennas have conical scans. The elevation antenna is tilted 5° above the horizontal plane.

A block diagram of the STOLAND navigation computation sequence is shown in Fig. 7. Navaid position data and body accelerations are transformed to components in the runway reference coordinate frame where they are filtered in separate x, y and z complementary filters. Again, the on board sensors used for navigation measurements are the TACAN receiver and the MODILS receiver, a body-mounted accelerometer package, the pitch, roll, and heading angles from the attitude and heading reference system, barometric altimeter, radio altimeter, and an airspeed sensor. The navigation subroutines develop estimates of position and velocity with respect to the local runway-fixed coordinate frame. In conjunction with air data, a wind vector is also estimated for use in the guidance computations. In case of navaid failure, the complementary filters are reconfigured for dead reckoning for a maximum of two minutes using air data and the last wind estimate.

Figure 8 presents a schematic diagram of the STOLAND complementary filter for estimating aircraft position $\hat{x}$ and velocity $\dot{\hat{x}}$ components along the runway centerline axis [2]. A similar filter configuration is used for computing $\hat{y}$ and $\dot{\hat{y}}$ normal to the runway. Note, in Fig. 8, that the acceleration measurement $\ddot{x}_r$, the airspeed measurement $\dot{x}_a$, and navaid position measurement x_r are input and blended in the filter to produce the smoothed estimates.

Figure 9 illustrates the complementary filter which is used for computing the vertical position $\hat{z}$ and velocity $\dot{\hat{z}}$ components. It has a slightly different configuration than do the x-y filters, and it uses altimeter and MODILS elevation derived altitude data for updating the estimated variables. A complete discussion of these filters is found in Ref. 2.

The roll, pitch and heading measurements from attitude gyros are used to compute the transformation matrix from aircraft body-fixed axes to the runway reference system.

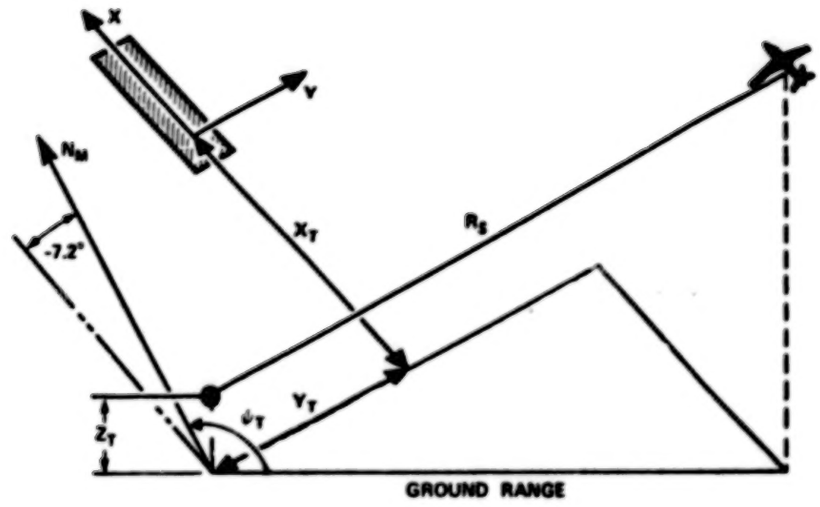


FIGURE 5.- DEFINITION OF TACAN COORDINATES IN THE RUNWAY COORDINATE FRAME [1]

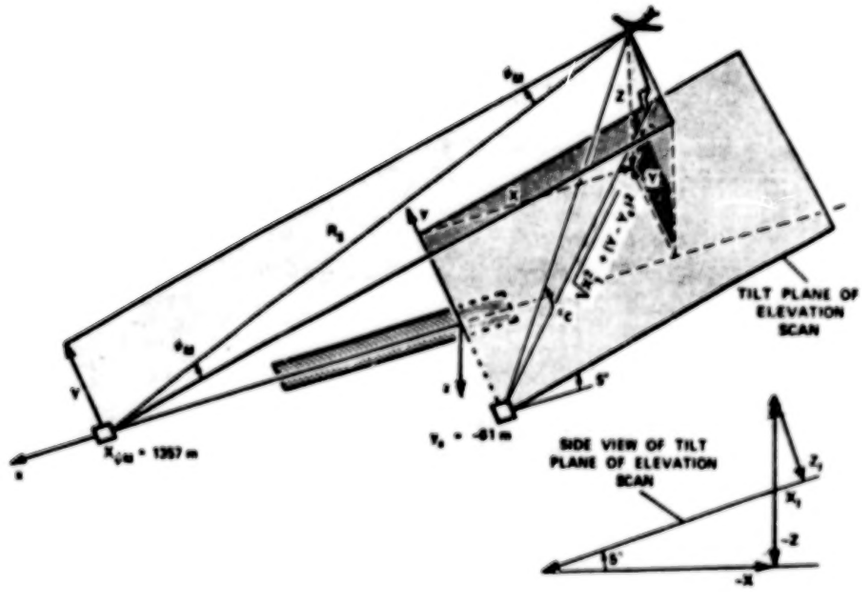


FIGURE 6.- GEOMETRY OF MODILS CONICAL SCAN ANTENNAS USING THE RUNWAY-ORIENTED COORDINATE SYSTEM [1]

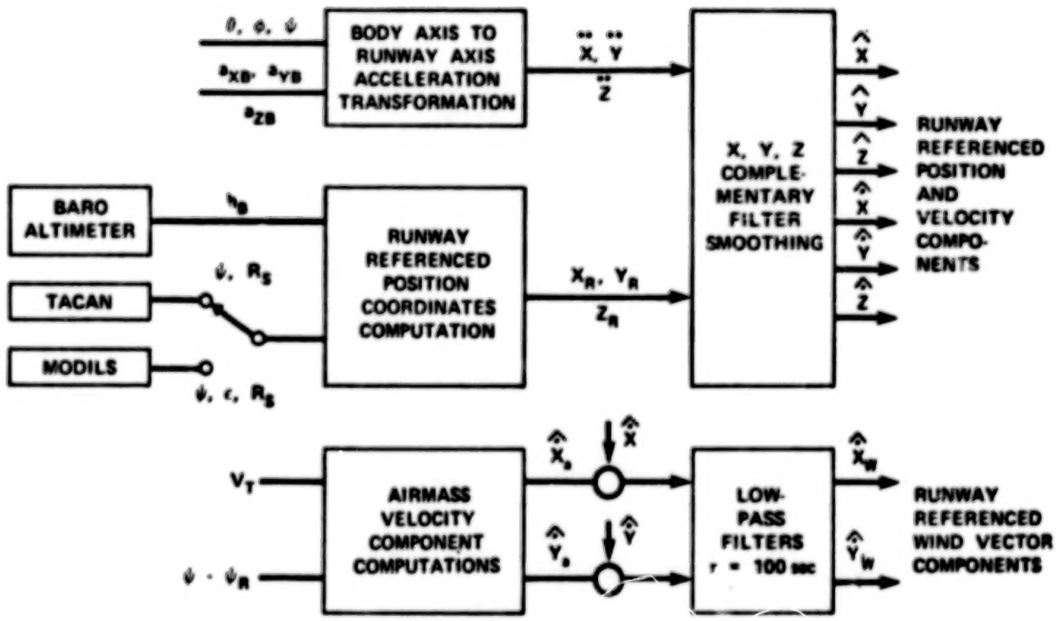
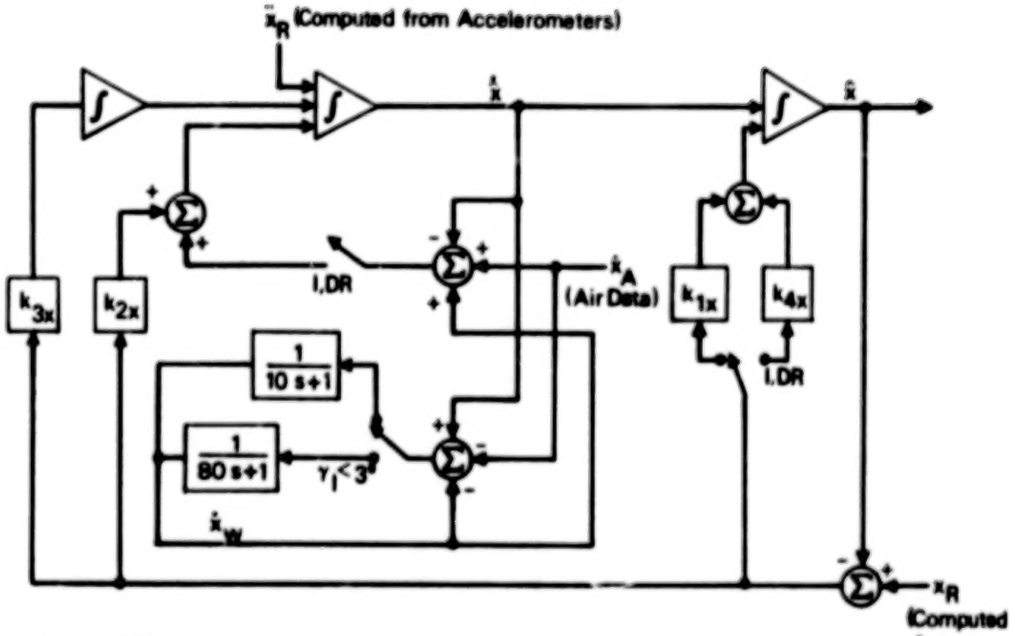


FIGURE 7.- BLOCK DIAGRAM OF STOLAND SYSTEM NAVIGATION COMPUTATIONS [1]



$k_{1x} = \sqrt{2k_{2x}}$

k_{2x} = RANGE-ANGLE DEPENDENT GAIN [2]

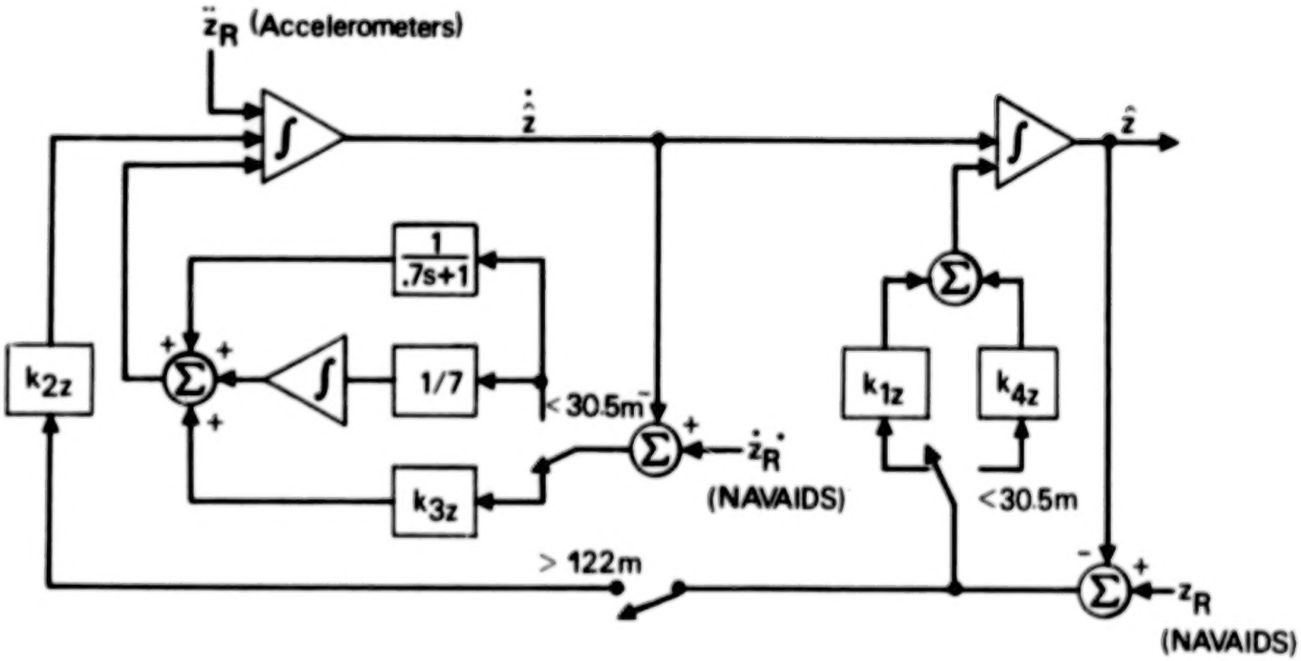
$k_{3x} = (k_{2x})^{3/2}/4$

$k_{4x} = 0.2$

I = INITIALIZATION (5 SEC AFTER TURN-ON)

DR = DEAD RECKONING

FIGURE 8.- SCHEMATIC OF STOLAND X-COMPLEMENTARY FILTER [2]



$$\dot{z}_R = [\, z_R(n) - z_R(n-1)]/\Delta t$$

$k_{1z} = 0.1$

$k_{2z} = 0.025$

$k_{3z} = 0.2$

$k_{4z} = 2.0$

FIGURE 9. SCHEMATIC OF STOLAND Z-COMPLEMENTARY FILTER [2]

This tranformation, in turn, is used with the accelerometer measurements to calculate the acceleration components $\ddot{x}_r$, $\ddot{y}_r$, and $\ddot{z}_r$ in the runway reference system. The raw navaid measurements are used to compute, for example, the position vector component x_r, in Fig. 8. The error between the measured position vector x_r and the estimated position vector component $\hat{x}_r$ is fed back with the gains k_{1x}, (or k_{4x}), k_{2x}, and k_{3x}, as illustrated in Fig. 8. Basically, there are eleven state variables in the three complementary filters--three components of each position, velocity and acceleration bias, plus two components of the estimated wind.

In the complementary filter implementation (see Ref. 2), the gains are either fixed or dependent on the range and bearing of the aircraft from the navaid station. Also, there is logic for: (a) the dead reckoning mode (when navaid measurements either are not available or are rejected), and (b) navaid selection. The complementary filter combines the inertial data with the navaid measurement data to give filtered velocity and position information.

The attitude reference system used in the STOLAND system develops moderately large errors in attitude during turns of the aircraft. This is the prime error source in the computed acceleration. Also, the accelerometers are not inertial grade equipment; the resulting accelerometer errors and error in the assumed gravity magnitude (which is assumed constant) also contribute to errors in the computed acceleration. The estimated acceleration biases are used to provide some compensation for these error sources. The filter gains are selected as a compromise in blending the low frequency acceleration errors and the high frequency noise errors of the raw navaid measurements. As is seen later, the STOLAND system complementary filter is able to provide adequate accuracy in position and velocity for automatic landing of the aircraft.

The Kalman Filter

Design considerations.- The selection of an appropriate Kalman filter configuration for testing in the STOLAND system involved the following considerations:

(1) The computer memory and real time available in the STOLAND system were quite restrictive. At the initiation of the study, it was believed that if the Kalman filter design required no more than

3000 words of memory or 20% of real time, then the 1819A computer could accommodate the mechanization.

(2) The Kalman filter design could be set up into two different forms, as follows:

(a) Actual calculation and updating of the covariance matrix in accordance with theory, and use of approximations consistent with the computer memory and computer time restrictions.

(b) Elimination of the covariance matrix computations by use of the theory to develop the feedback structure, and use of piecewise constant (or state dependent) gains.

Because this was a research investigation, either approach could have been chosen. Approach (b) would have resulted in a less demanding filter from memory and real time considerations. It would provide a configuration very nearly the same as the complementary filter. However, it would be very specialized to only the terminal area and landing phase of flight. Approach (a) was selected for the investigation because proof of its applicability for landing usage would prove its applicability for all phases of flight. The discrete square root form of the Kalman filter [10] was used because of its mechanization advantages.

(3) The accuracy of the STOLAND system inertial measurements degrades substantially in going from level flight into turns. A good mathematical model of these error characteristics was not available, so the filter design needed to be somewhat ad hoc in the manner of obtaining compensation for these characteristics.

(4) As the aircraft passes from enroute to the landing phase, the accuracy of the navigation aids (TACAN to MODILS) was expected to improve substantially. In the present study, TACAN range and bearing and barometric altitude were used in the terminal area mode. Transition from using the TACAN measurements to more accurate MODILS range and azimuth data occurred prior to the turn onto the final approach. Transition from barometric altitude to altitude determined from MODILS elevation data occurred on the final approach. Transition from MODILS elevation to radio altitude occurred below 30 m (100 ft).

The Kalman filter had to be designed to provide smooth transition in going between these different sources of position measurement information.

The Kalman filter used in this study was developed in stages, as is discussed in Refs. 5 and 6. This allowed the above considerations to be incorporated into the filter design which was flight tested.

System description.- The Kalman filter, as mechanized within the STOLAND system, is depicted in Fig. 10. As can be seen, the added blocks include the navigation equations, the x-y filter, the z filter, and the smoothing logic.

The filter equations were originally developed for the horizontal plane [5] (x-y filter) and then for the vertical direction (z filter) [6]. It proved to be efficient to maintain this decoupling in the mechanization.

The equations which were used to develop the Kalman filter design are presented in detail in Appendix A. These are summarized in Table 2. The equations were divided and mechanized into three priority levels as shown in this table. They are referred to as Foreground, First Level Background, and Second Level Background equations. An explanation of how the available real time of the STOLAND system is divided to mechanize the three priority levels is given in Appendix B.

The Foreground (Navigation) equations are used to integrate the accelerometer readings at a high rate (20 Hz). The First Level Background equations accumulate and preprocess the position and air data measurements at one-half (10 Hz) the integration rate.

There was not sufficient real time available to execute the entire Kalman filter equations at this speed. Thus, the majority of the Kalman filter computations were distributed over the remaining real time available during a 1.5 sec period (0.667 Hz). These Second Level Background computations are listed in Table 2, and they are explained in Appendix A.

At the end of every 1.5 sec period, there is a discrete update to the estimated state variables based on processing external measurements by the Kalman filter. These updates could cause undesirable jumps in the estimates, from a usage (display, control) point of view. Thus, the smoothing logic shown in Fig. 10 was added to take out the abrupt changes. This logic is explained in Appendix A. Further details on interfacing the Kalman filter with the STOLAND system and initialization of the filter are also presented in Appendix A.

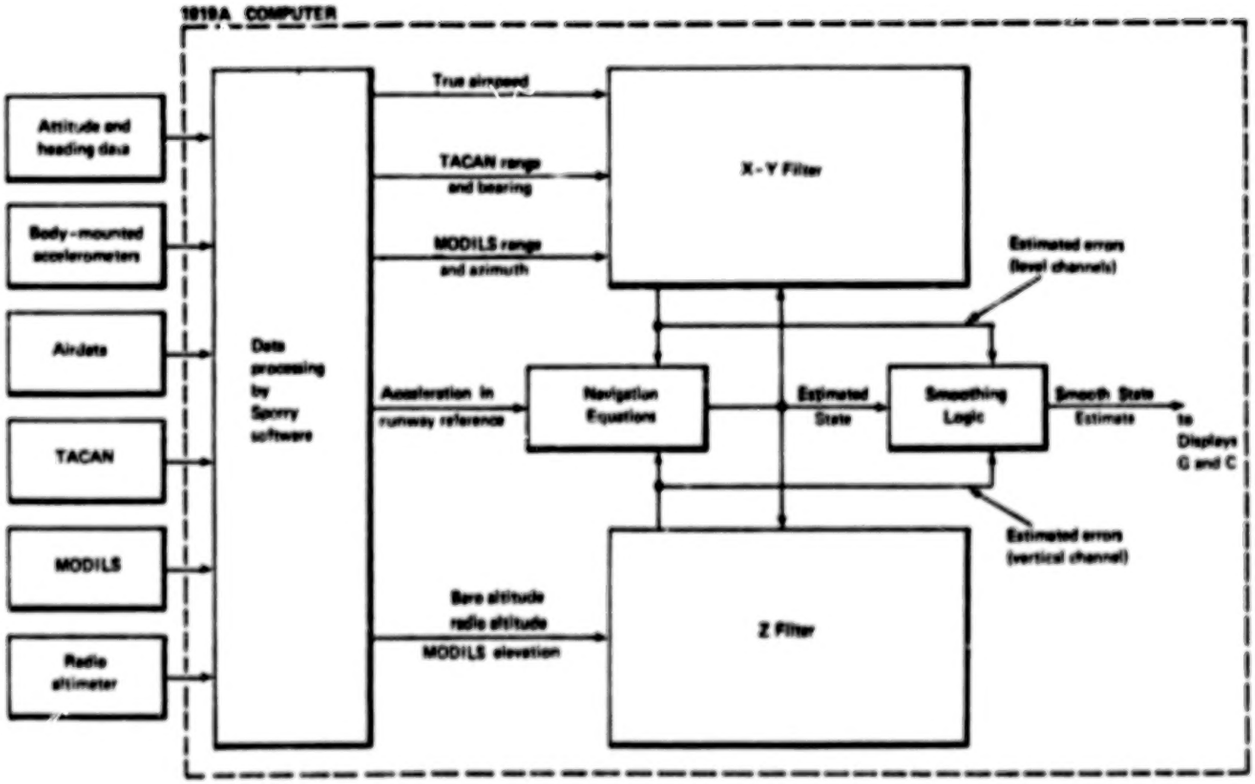


FIGURE 10.- BLOCK DIAGRAM OF THE KALMAN FILTER MECHANIZATION WITHIN THE STOLAND 1819A COMPUTER.

TABLE 2. - SUMMARY OF MECHANIZED KALMAN FILTER EQUATIONS

THE EQUATIONS SOLVED IN THE 1819A COMPUTER ARE SUMMARIZED BELOW

RATE	EQUATIONS	EXPLANATION
20 Hz (Foreground)	(Δ = 0.05 sec). Update state equations (Navigation equations)	
	$\dot{x}^i(t+\Delta) = \dot{x}^i(t) + [\ddot{a}_s^{i-3} + \ddot{x}^{i+3}(t)]\Delta$	update velocities i = 4-6
	$\dot{x}^i(t+\Delta) = \dot{x}^i(t) + [\dot{x}^{i+3}(t+\Delta) + \dot{x}^{i+3}(t)]\Delta/2$	update position i = 1-3
	$\dot{x}^i(t+\Delta) = \dot{x}^i(t)$	update states i = 7-14
10 Hz (First Level Background)	(Δ = 0.1 sec). Measurement preprocessing; executed for all valid measurements (k = 1-6)	
	$\Delta y^k = Y^k - \hat{Y}^k$	compute residual
	$y_s^k = y_s^k + \Delta y^k$	residual sum
	$H^k = \nabla_x \hat{Y}^k$	compute measurement gradient (partial)
	$H_m^k = H^k \Phi$	refer partial to beginning of 0.667 Hz cycle
	$H_s^k = H_s^k + H_m^k$	partial sum
0.667 Hz (Second Level Background)	(Δ = 1.5 sec). Update incremental state and covariance matrix using Potter's algorithm: Executed for each of the five measurements in effect (MODILS or TACAN, air data, and altitude)	
	$S_k = H_s^k W W^T (H_s^k)^T + Q_s^k$	variance in residual
	$d\hat{x} = d\hat{x} + W W^T (H_s^k)^T [y_s^k - H_s^k d\hat{x}]/S_k$	update incremental state vector
	$W^T = W^T - \dfrac{W^T (H_s^k)^T H_s^k W W^T}{[S_k(1+\sqrt{Q_s^k/S_k})]}$	update square root covariance matrix
	Add incremental state to state estimates (i = 1-14)	
	$dx(t) = \Phi dx$	refer state change to current time
	$x^i = x^i + dx^i$	update states
	$dx^i = 0$	
	Update W^T with forcing functions	
	$W^T(t+\Delta) = \begin{bmatrix} W^T(t)\Phi \\ U^T \Phi_u^T \end{bmatrix}$	
	$W^T(t+\Delta)$ = upper triangular form using Householder's algorithm.	

One of the objectives of this study was to compare the performance of the Kalman filter to that obtained from the complementary filters which are a part of the regular STOLAND software. It is useful to note the differences of the two mechanizations at this time. In Figs. 8 and 9, the complementary filters are depicted as being mechanized in a continuous fashion. This is essentially correct in that the integrations are mechanized digitally in the STOLAND software with an update rate of 20 Hz.

The Kalman filter can also be described with continuous equations [11]. Then, these equations can also be placed in analog form, as is shown for the complementary filters. This is done for the z filter in Fig. 11. The basic form is essentially the same, as is discussed in Ref. 12. The primary differences are as follows:

(1) The complementary filters have position dependent or fixed gains (e.g., k_{1x}, k_{2x}, k_{3x}, k_{4x} in Fig. 8; k_{1z}, k_{2z}, k_{3z}, k_{4z} in Fig. 9). The Kalman filter has continuously changing gains (e.g., k_{1z}, k_{2z}, k_{3z}, k_{4z} in Fig. 11). These gains are computed to be optimal using the square root covariance matrix. They are defined by Eqs. (A.11) of Appendix A.

(2) The complementary filters are essentially decoupled (e.g., x, y, and z filters). The Kalman filter takes advantage of natural coupling and has cross-coupling gains. This provides better accuracy at the cost of complexity.

(3) The Kalman filter provides a systematic way of computing additional information which improves the accuracy of the output. This information is in the form of additional state variables which can be used to remove measurement errors. For this study, the Kalman filter provided three components of position, velocity, and acceleration bias and two components of the horizontal wind, as did the complementary filters. The Kalman filter also provided estimates of TACAN range and bearing measurement bias and baro-altimeter bias.

Each of the above points allow the Kalman filter to provide greater accuracy. The cost is greater mechanization complexity.

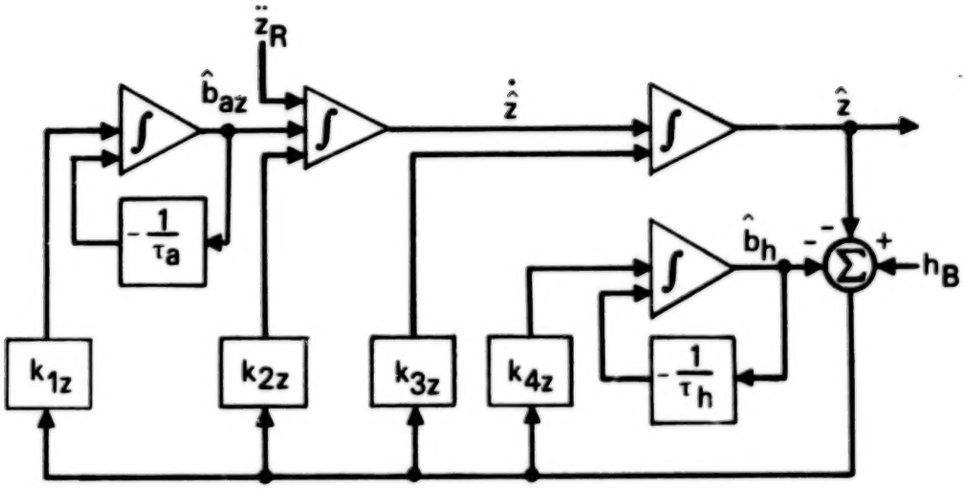


FIGURE 11.- SCHEMATIC ANALOG FORM OF KALMAN FILTER FOR VERTICAL DIRECTION.

Comparison of Mechanization Requirements

The three-axis Kalman filter mechanization used in the flight tests required approximately 3000 words of memory and 18% real time availability of the 1819A computer. Minor modifications in the Sperry STOLAND display logic were made in order to obtain the required time for the Kalman filter. As was previously mentioned, the complementary filter requires 1,066 words of memory and 12% real time availability of the 1819A computer. These demands are not so large as to preclude the use of the Kalman filter in the operational sense.

BLANK PAGE

IV

FLIGHT TEST RESULTS

To test the Kalman filter navigation system and to compare its performance with that of the STOLAND complementary filters, five different landing approaches (flight segments) were made using the system on Dec. 10, 1976. This chapter first describes the flight test facilities and data processing procedures, the data recorded during the tests, and the flight profiles flown. Then, the results of these tests are summarized.

Flight Test Description

The five test flight segments were made at the NASA Crows Landing NALF test facility near Modesto, Calif. This facility has a radar tracking system consisting of two modified Nike-Hercules radars. The modification provides improved position resolution through the use of 19-bit range and angle digital shaft encoders. The flights were made using the NASA Ames Twin Otter aircraft. A transponder on board the aircraft was used to improve the radar angular tracking.

A data recording system on board the aircraft was used to obtain the outputs from the STOLAND complementary and Kalman filters plus the raw measurements from the on board instruments. The instrument measurements included the TACAN bearing and range; the MODILS azimuth, elevation, and range; the true airspeed; the barometric and radio altitude; the x, y, and z body-mounted accelerometer outputs; and the roll, pitch, and heading angles as measured by the vertical and directional gyros. Also, these data measurements were directly telemetered to the ground and digitally recorded. Clocks on the aircraft and at the ground facility were initially synchronized so that the data recorded on the aircraft could also be directly correlated with that obtained from the ground-based radars.

The radar position measurements were combined with the aircraft accelerometer measurements in complementary filters implemented in the post-flight data processing program. This filtering procedure was used to smooth the radar measurements and to derive good estimates of the aircraft's actual position and velocity throughout each flight segment. These smoothed, radar derived trajectories were then used as the standard with which to assess the performance of the Kalman and complementary filters.

The differences between the three components of position (x,y,z) and velocity ($\dot{x},\dot{y},\dot{z}$) derived from the on board Kalman and complementary filters and the same components derived from radar were computed and compared. An assessment of this data was then made to evaluate the two filters' performances.

The five segments flown consisted of a series of touch-and-go profiles. These profiles are shown in Figs. 12-16. These figures show each trajectory from a horizontal view (x-y plane) and a runway planar view (x-z plane). Discrete events, such as the transition point from TACAN range/bearing to MODILS range/azimuth measurements, are indicated on these profiles. The record of the navaid measurements (airspeed, altitude, MODILS range, bearing, and elevation) and inertial measurements (accelerometer and attitude gyro) for each of these profiles are depicted in Figs. 17-21 as a function of time. Figures 12-21 can be used to determine the unique features of each flight segment which affect the performance of the filters.

The first flight segment, shown in Fig. 12, consisted of a typical approach, landing, and rollout sequence. In this test, the initial navigation information came from TACAN range and bearing measurements and the baro-altimeter. (Information from the baro-altimeter remained valid throughout this and all other flight segments.) At 130 sec past the initial point, the MODILS range and azimuth data became valid. (Both hardware discretes (range and azimuth) show valid data, and the azimuth angle becomes less than 20°.) Thus, MODILS range and azimuth measurements replaced the TACAN data beyond this point for input to the x-y filter. At 202 sec past the initial point, the MODILS elevation information was accepted by the z filter. (The criteria for its acceptance is that the MODILS elevation valid flag must be true, the aircraft x-position (in the runway reference frame) must be more positive than -10,856 m (-35,616 ft), and the aircraft heading angle with respect to the runway must be within $\pm$20°.) At 328 sec past the initial point, the radar altitude became less than 30.5 m (100 ft). Thus, beyond this point, it was used instead of the MODILS elevation measurements for input to the z filter. Touchdown occurred at 341 sec past the initial point.

The second flight segment, depicted in Fig. 13, began with the aircraft having a 49 m (160 ft) altitude shortly after take-off. The aircraft climbed out to about 457 m (1,500 ft) altitude, and it was positioned to begin a typical 45° approach leg. Then, an approach and landing sequence was executed similar to that of the first segment (Fig. 12). During the landing portion, the aircraft descended to about 12.2 m (40 ft) altitude after which it climbed out again. The

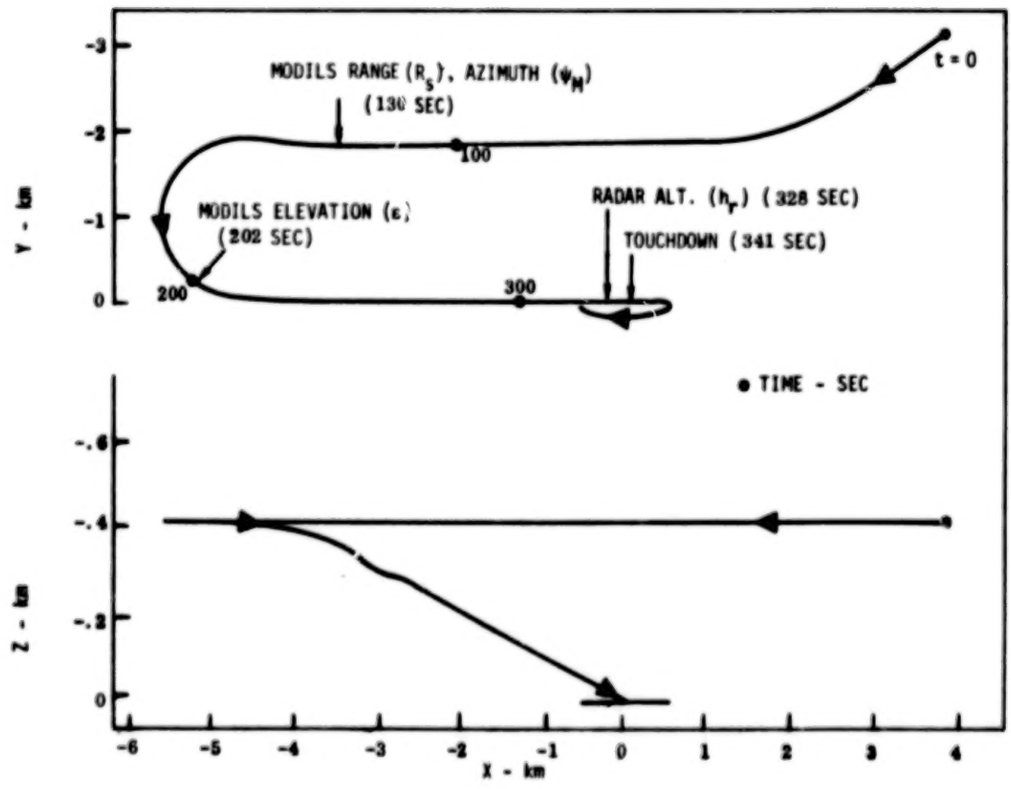


FIGURE 12.- PROFILE OF FIRST FLIGHT SEGMENT.

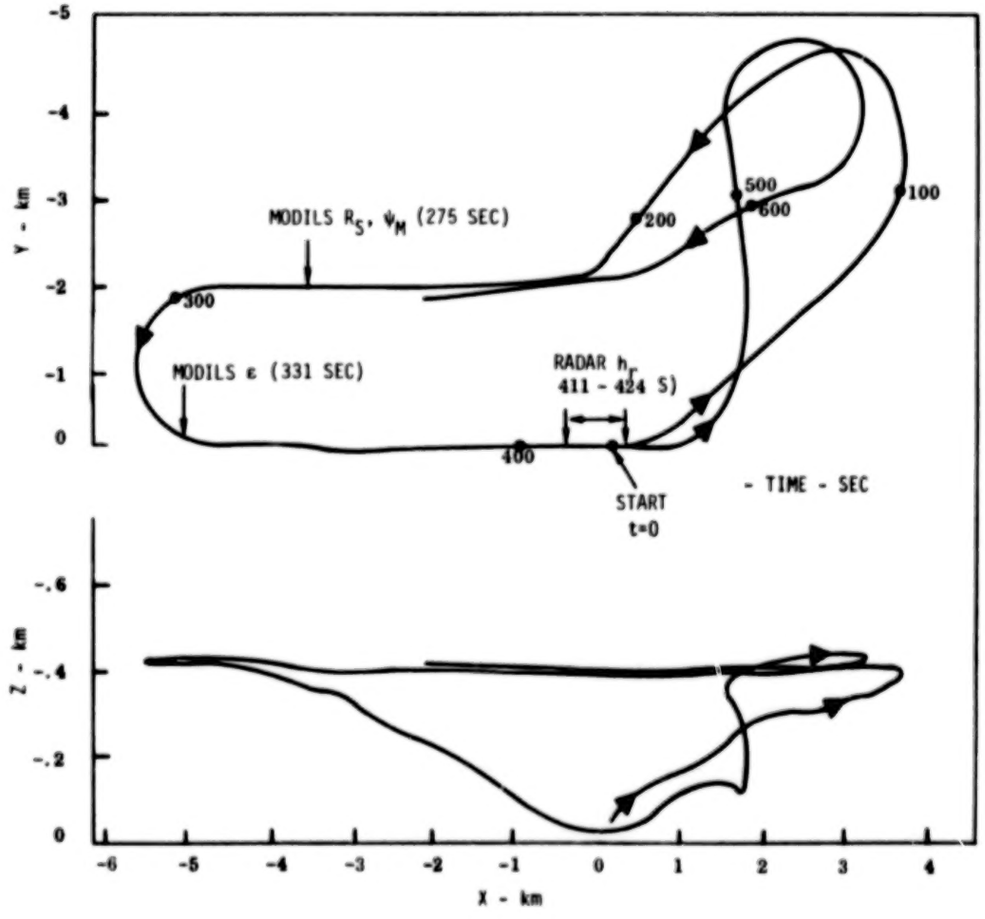


FIGURE 13.- PROFILE OF SECOND FLIGHT SEGMENT.

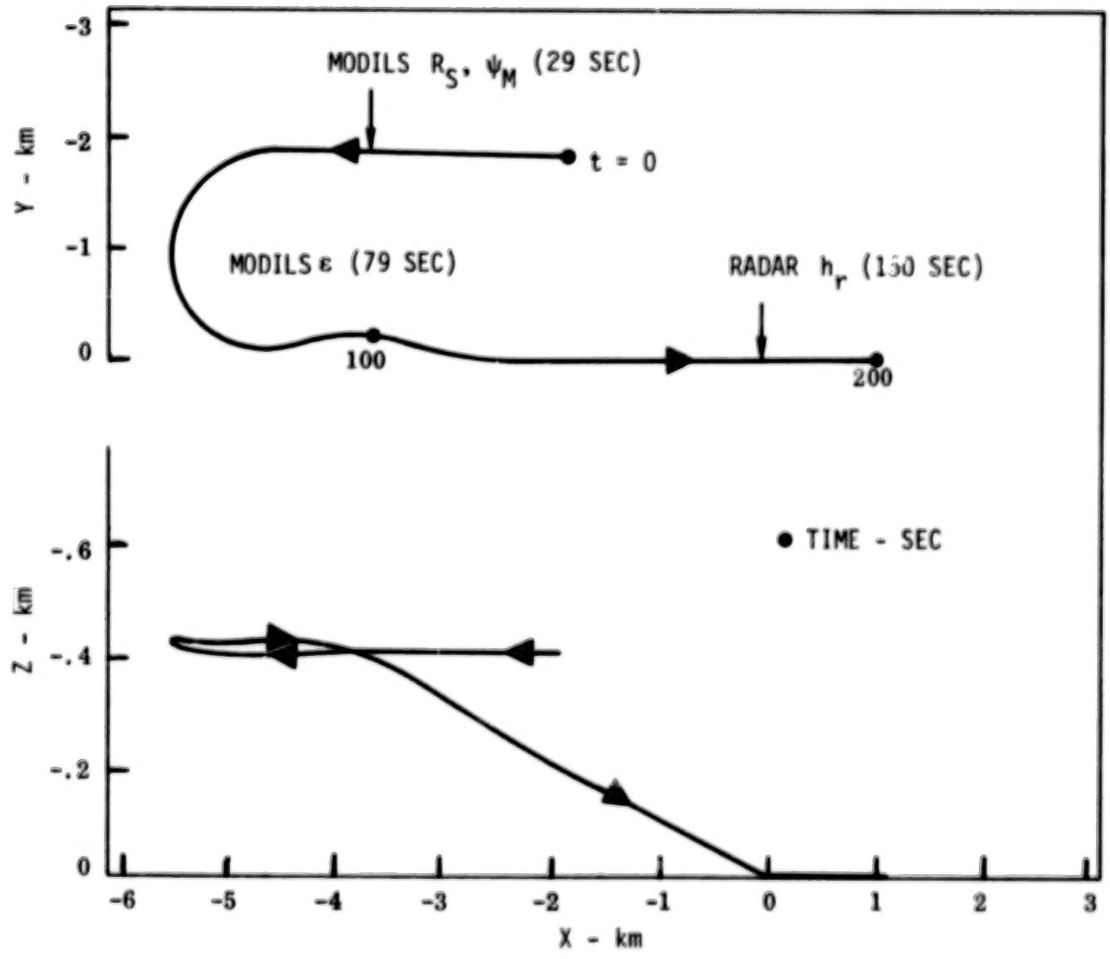


FIGURE 14.- PROFILE OF THIRD FLIGHT SEGMENT.

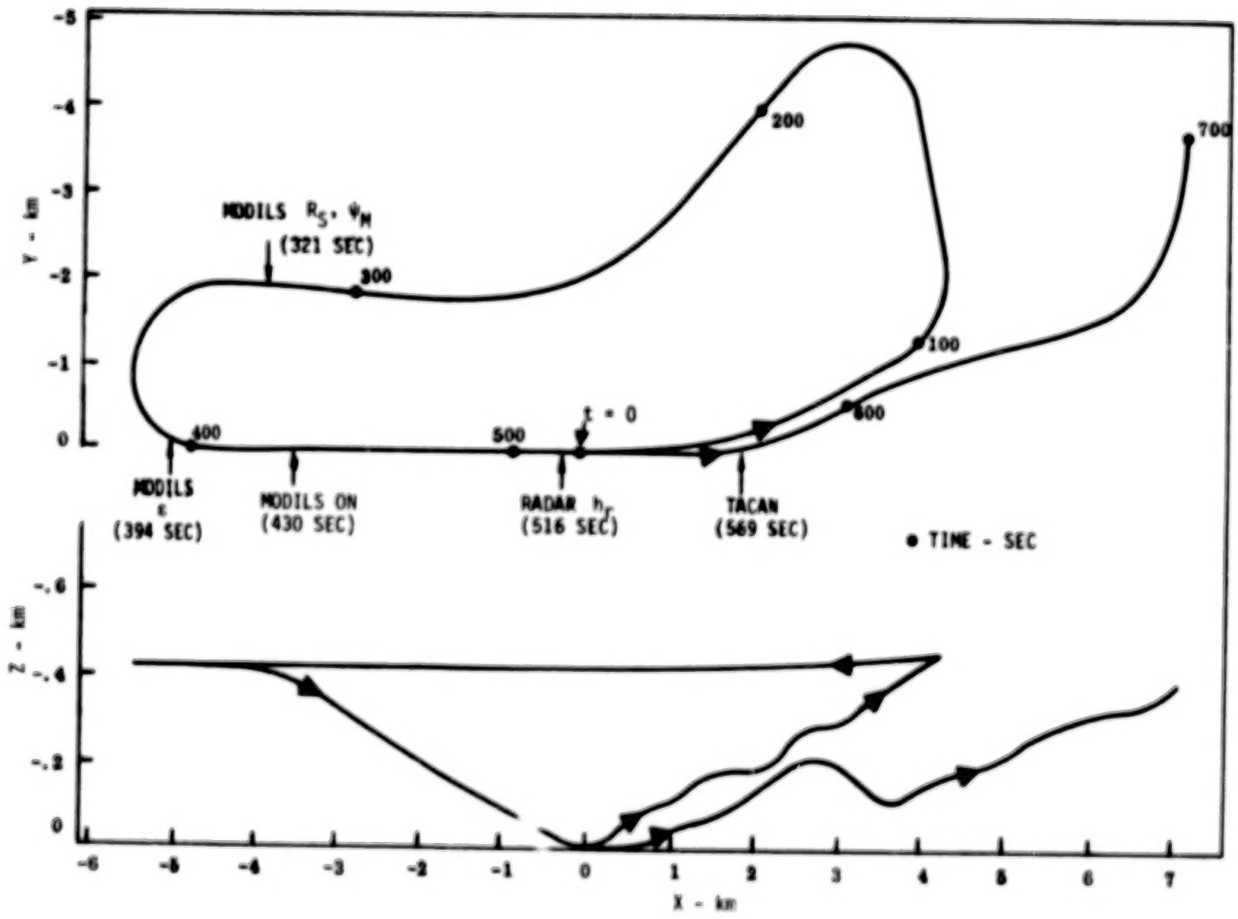


FIGURE 15.- PROFILE OF FOURTH FLIGHT SEGMENT.

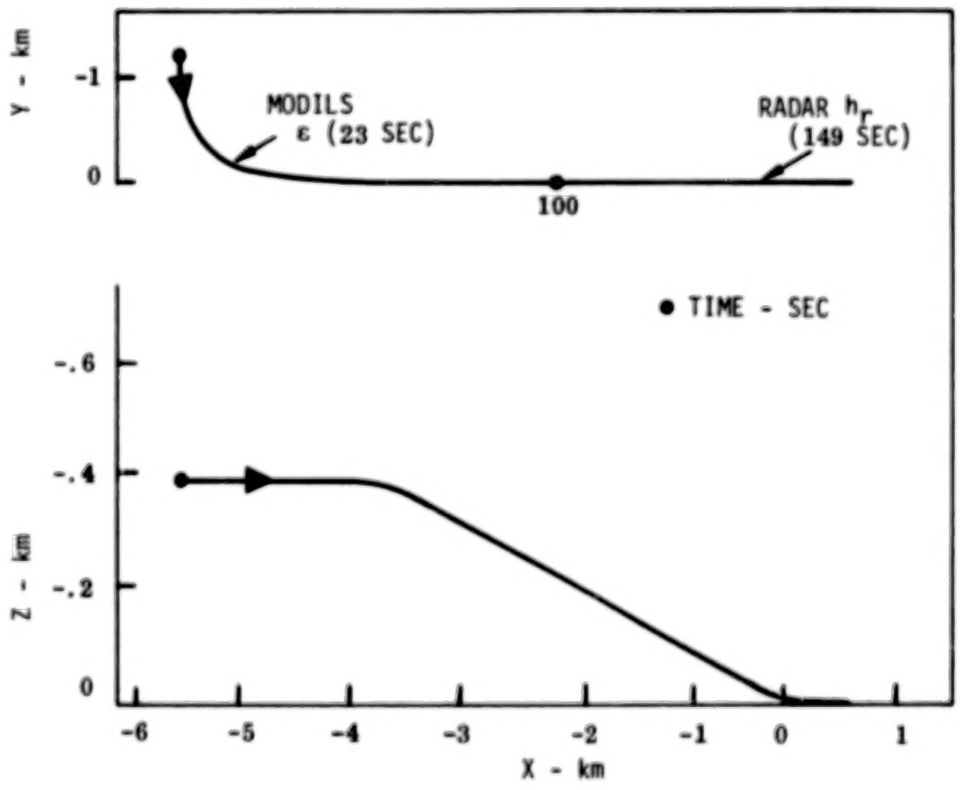


FIGURE 16.- PROFILE OF FIFTH FLIGHT SEGMENT.

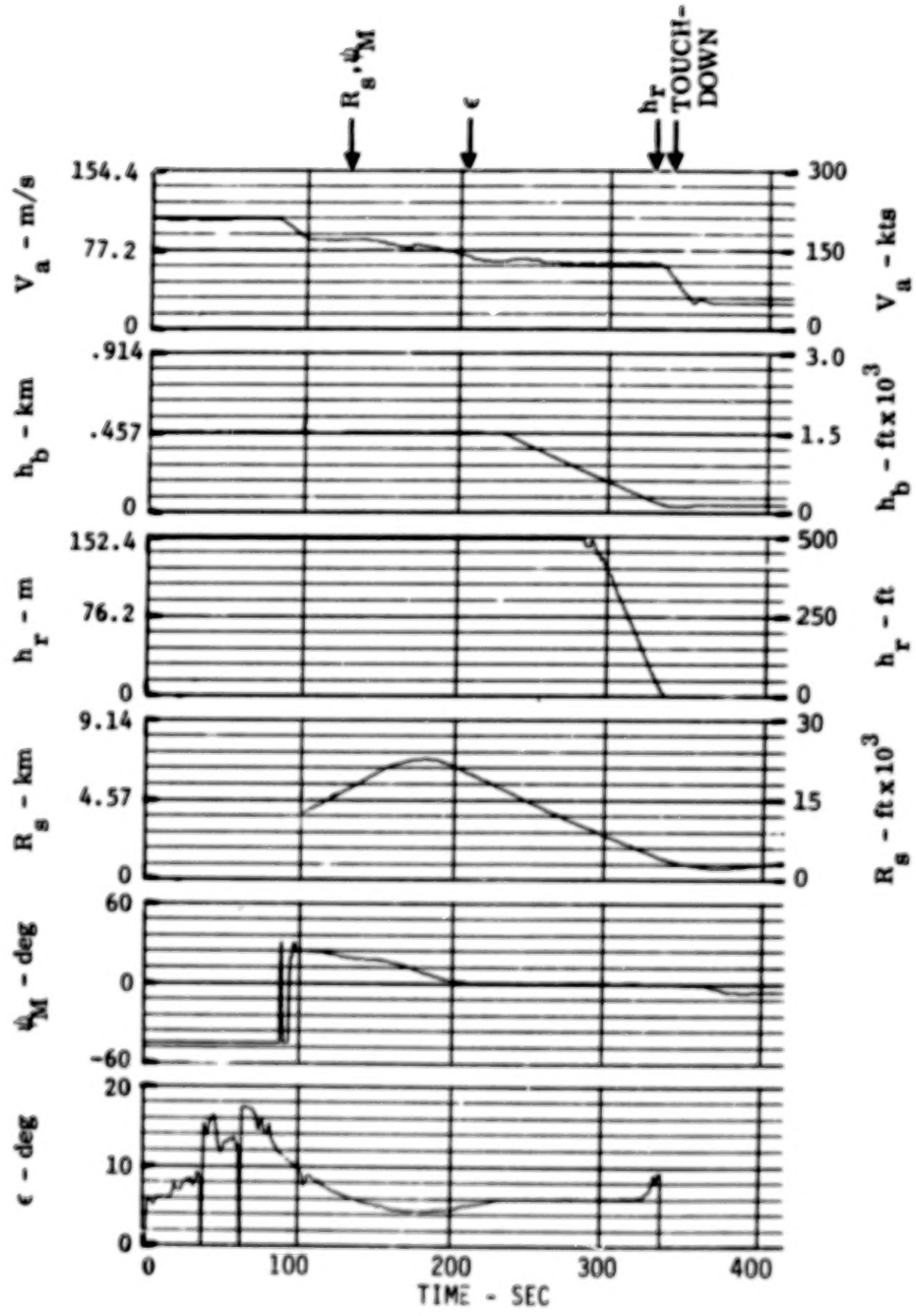


FIGURE 17a.- NAVAID MEASUREMENTS FOR FIRST FLIGHT SEGMENT

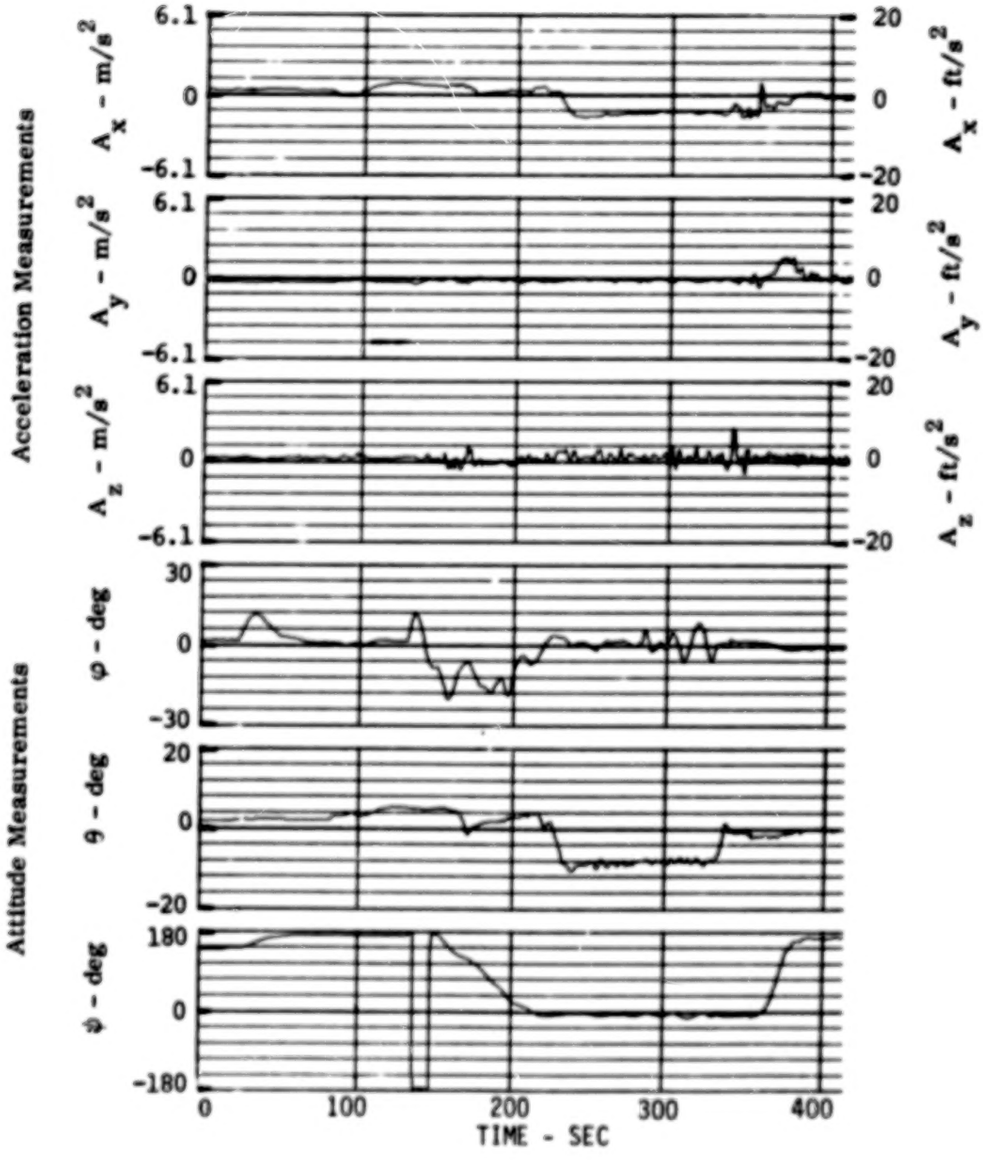


FIGURE 17b.- INERTIAL MEASUREMENTS FOR FIRST FLIGHT SEGMENT

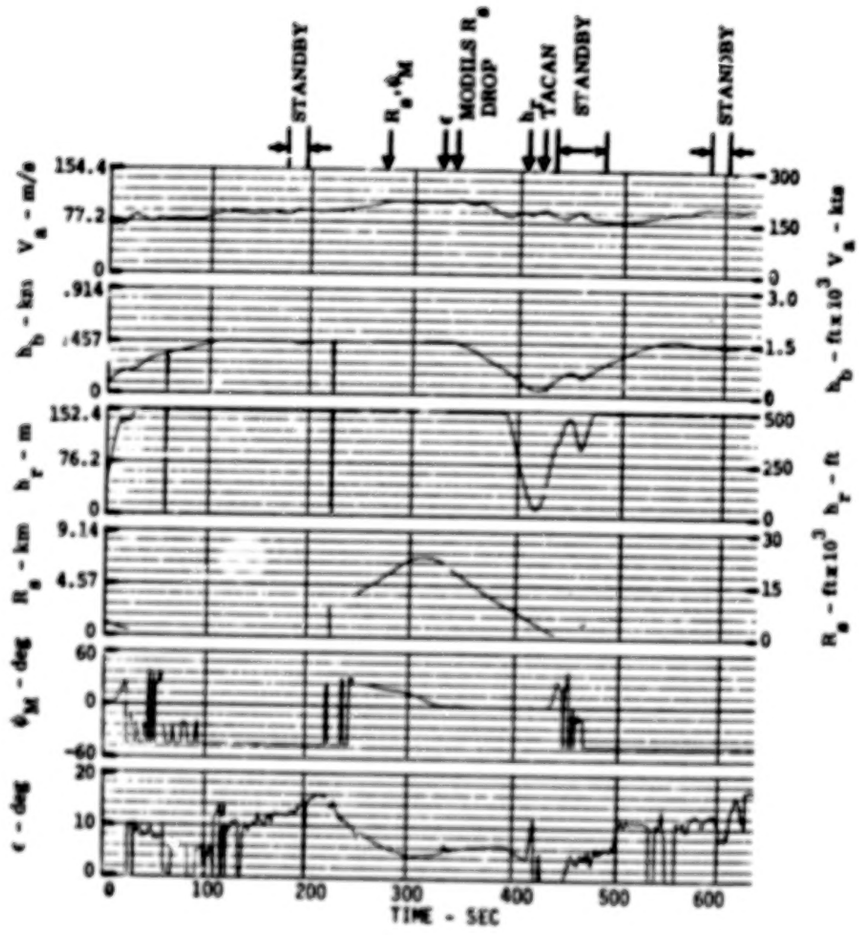


FIGURE 18a.- NAVAID MEASUREMENTS FOR SECOND FLIGHT SEGMENT

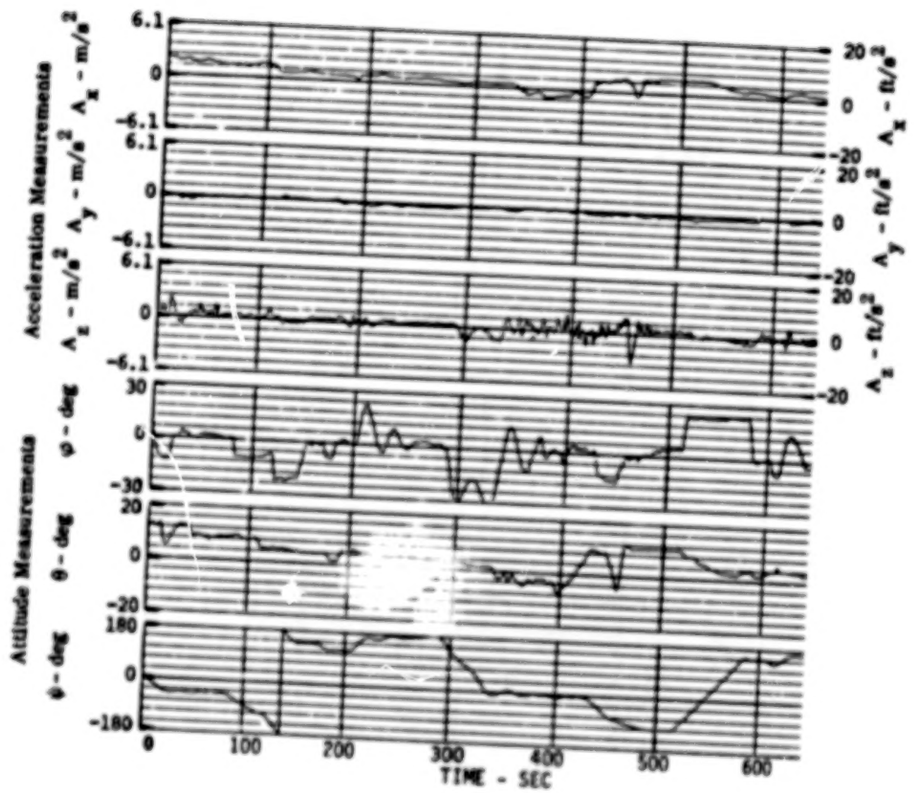


FIGURE 18b.- INERTIAL MEASUREMENTS FOR SECOND FLIGHT SEGMENT

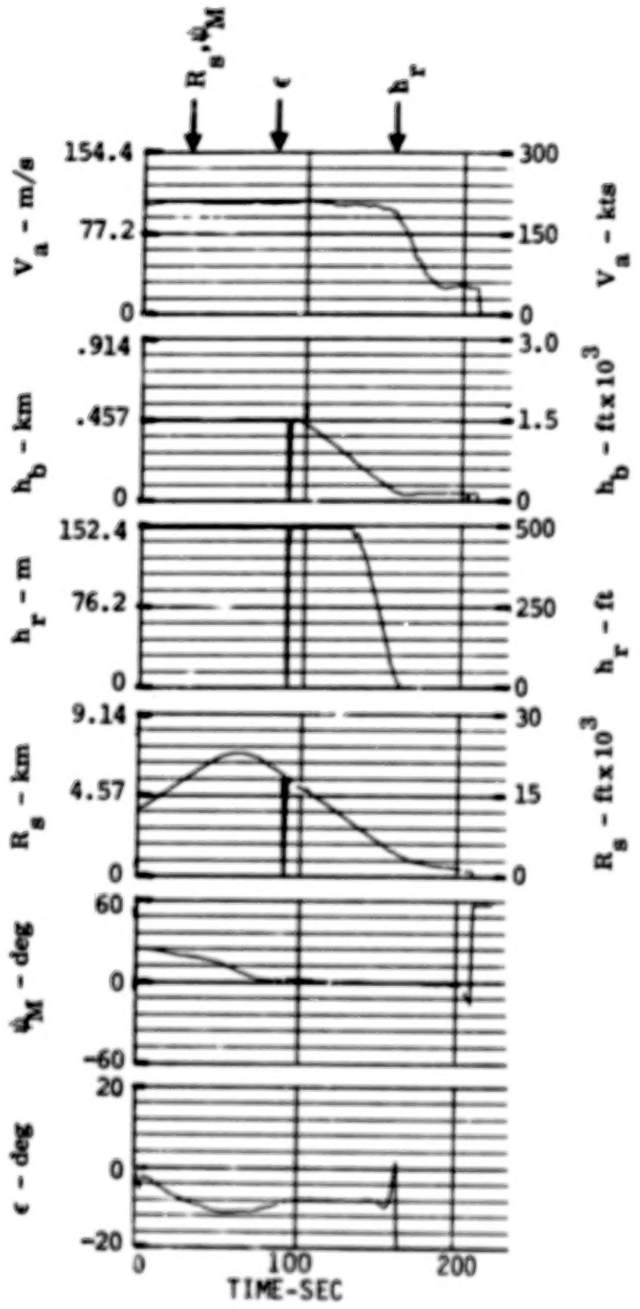


FIGURE 19a.- NAVAID MEASUREMENTS FOR THIRD FLIGHT SEGMENT.

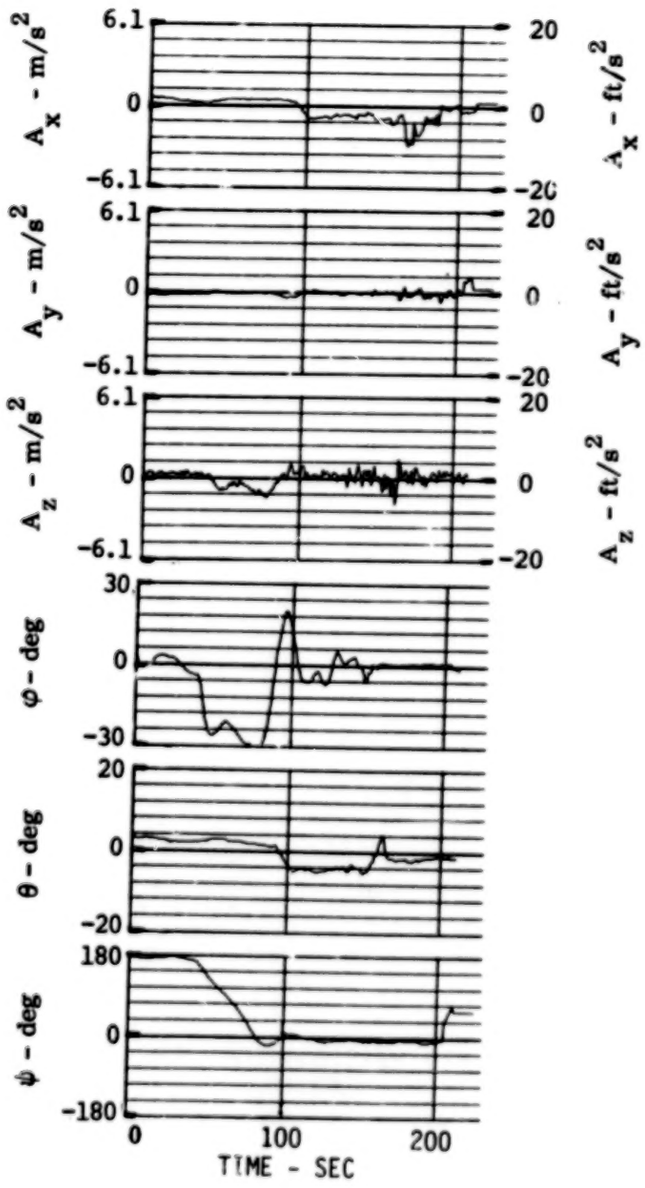


FIGURE 19b.- INERTIAL MEASUREMENTS FOR THIRD FLIGHT SEGMENT.

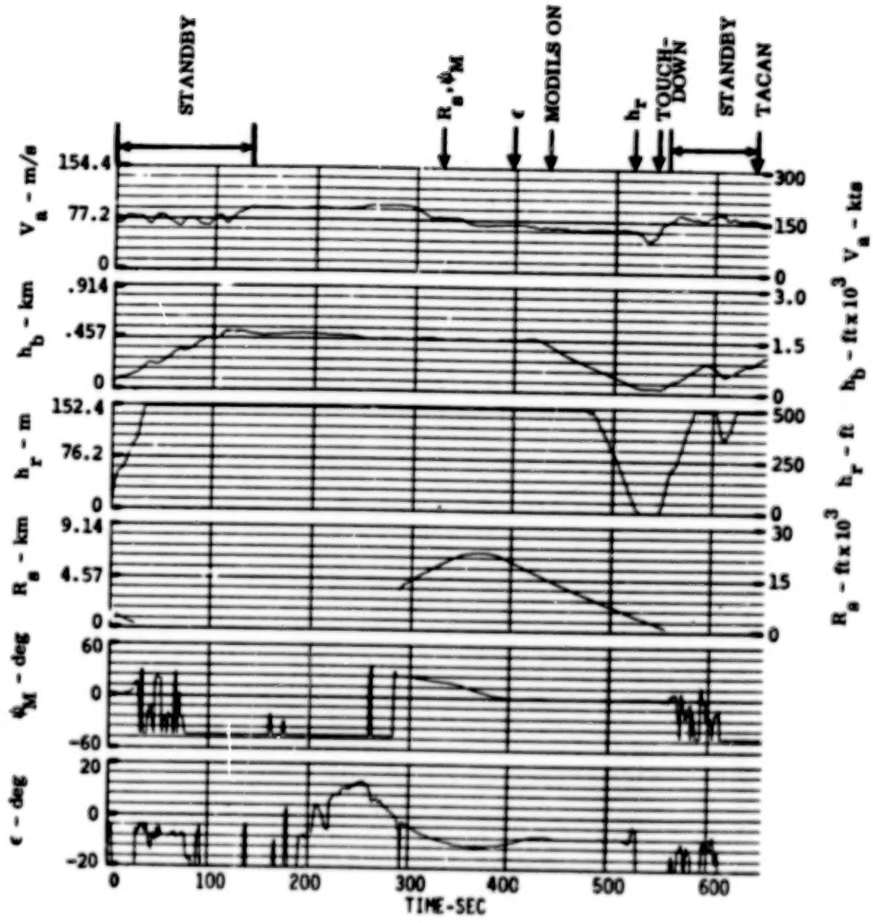


FIGURE 20a.- NAVAID MEASUREMENTS FOR FOURTH FLIGHT SEGMENT.

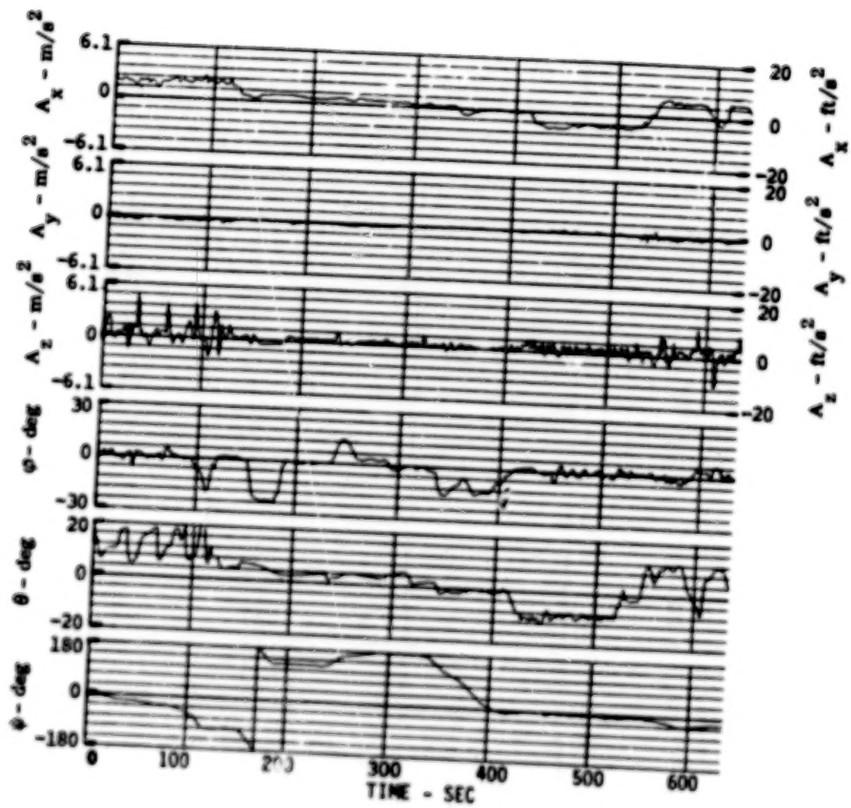


FIGURE 20b.- INERTIAL MEASUREMENTS FOR FOURTH FLIGHT SEGMENT.

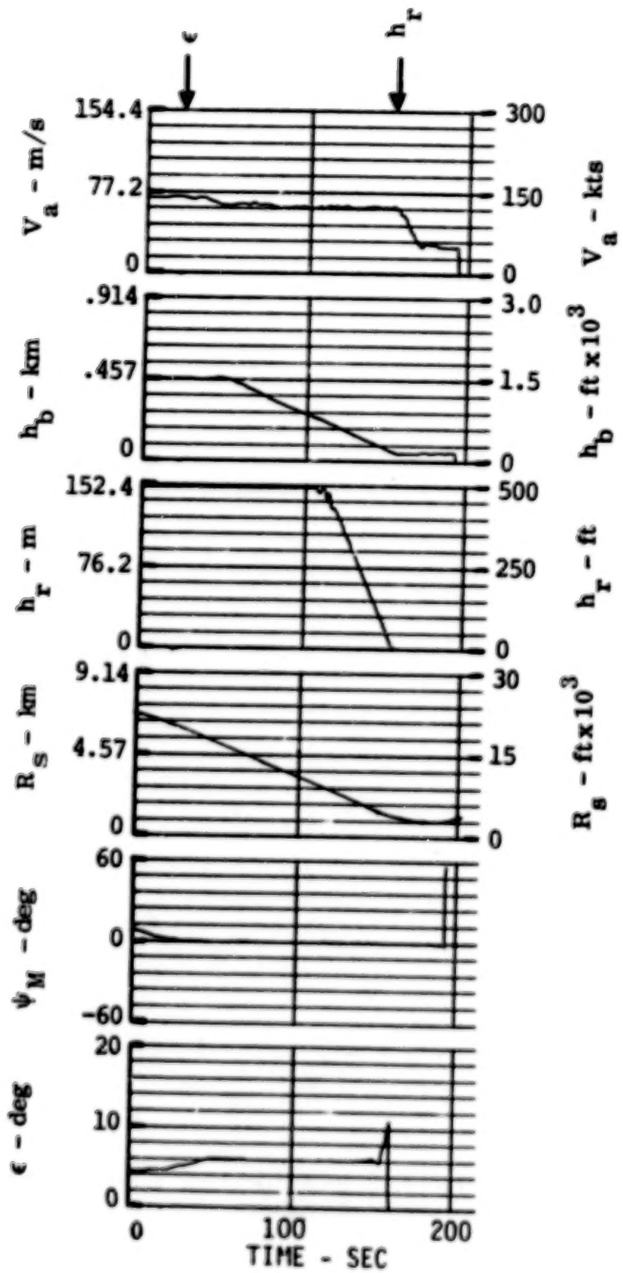


FIGURE 21a.- NAVAID MEASUREMENTS FOR FIFTH FLIGHT SEGMENT.

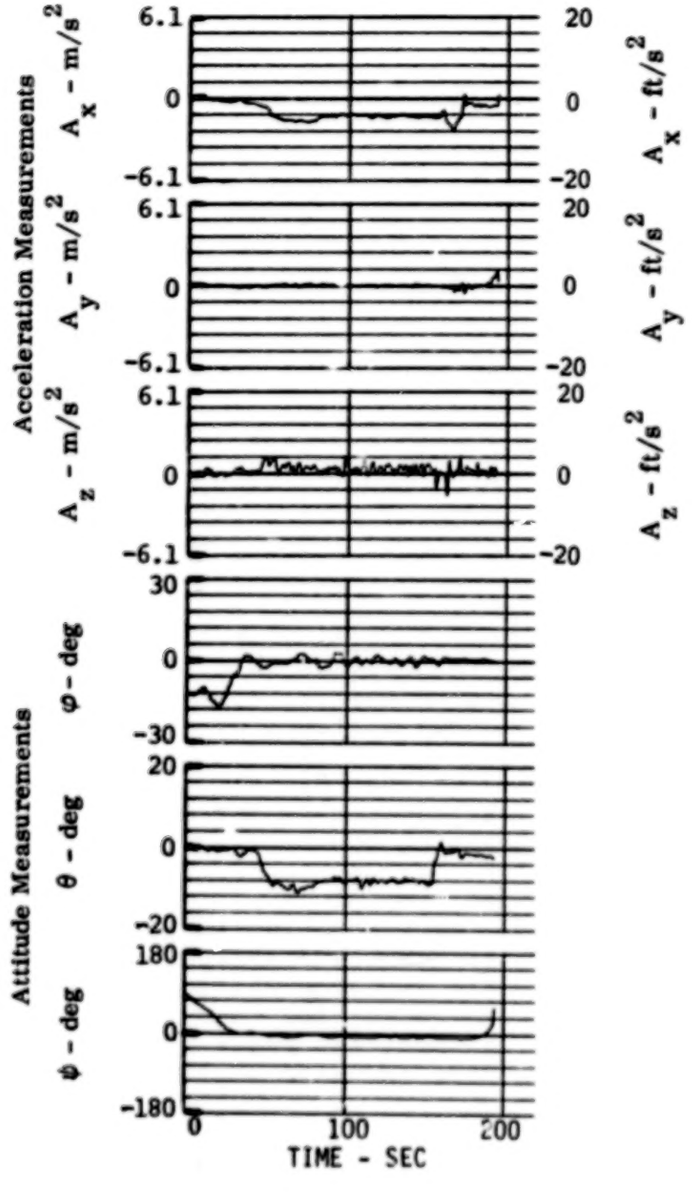


FIGURE 21b.- INERTIAL MEASUREMENTS FOR FIFTH FLIGHT SEGMENT.

final portion of the trajectory consisted of a maneuver that put the aircraft again on the downwind leg of the approach. The MODILS range and azimuth measurements became valid at 275 sec after initiation. The MODILS elevation data became valid at 331 sec. Between 333 sec and 337 sec, the MODILS range data were invalid (data dropout). The radar altimeter data were used beginning at 411 sec (altitude below 30.5 m (100 ft)). The x-y filter switched back to TACAN measurements at 421 sec. The TACAN data remained valid for use throughout the remainder of this flight segment.

The third flight segment, shown in Fig. 14, began on the downwind leg of the approach and ended during the rollout phase. MODILS range and azimuth data became valid at 29 sec past the initial point. MODILS elevation data became valid at 79 sec. Radar altitude started being used at 155 sec. Touchdown occurred at 168 sec past the initial point.

The fourth flight segment, shown in Fig. 15, began during a climbout at about 15.2 m (50 ft) altitude. This segment consisted of climbout to 457 m (1,500 ft), positioning for a 45° approach leg, normal downwind, base, and final approach legs, landing, take-off, and climbout again to 457 m (1,500 ft). The segment began with the filter using TACAN data. MODILS range and azimuth data became valid at 321 sec. MODILS elevation data became valid at 394 sec. However, the navaid selection was not switched by the pilot to MODILS until 430 sec at which time the MODILS data were used. Radar altimeter data were used between 516 sec and 551 sec. Touchdown was at 533 sec. Take-off was at 544 sec. The filters began using TACAN data again at 569 sec.

The fifth flight segment, depicted in Fig. 16, consisted of a base leg, final approach, and landing. This segment began with the MODILS range and azimuth data being used by the x-y filter. At 23 sec past the initial point, the MODILS elevation data became valid, and they were used for altitude computations. At 149 sec, the radar altimeter data were accepted, and touchdown occurred at 161 sec.

Flight Test Results

Before the flight tests were conducted, two different outcomes were expected. First, it was thought that the general navigation accuracy would improve from both filters in going from TACAN data to MODILS data. This was because a TACAN system is designed to have accuracies adequate for enroute navigation usage, and MODILS is intended to provide precision terminal area navigation information.

Second, it was expected that the Kalman filter would provide superior navigation information to that of the complementary filter. This was because the Kalman filter was configured to estimate and compensate for TACAN range and bearing bias and baro-altitude bias. In the previous simulation study of the x-y filter [5], this proved to be the case. However, these simulation results were based on the assumption that large TACAN and baro-altitude bias errors were indeed present.

Figures 22a-26a show the differences (residuals) between the three components of position and velocity, as obtained from radar measurement-derived estimates, and those components estimated by the Kalman filter, for each of the five flight segments. For example, for the x position component,

$$x_{res} = x_{radar} - x_{filter} \tag{1}$$

Figures 22b-26b show the corresponding residuals which were computed using the complementary filter estimates. Thus, the residuals of the two filters can be directly compared to get an indication of relative performance of the two methods.

Figures 22a and 22b show the residuals from the first flight segment which consisted of a single approach and landing trajectory. The MODILS range and azimuth data became valid at 130 sec, the MODILS elevation data were valid at 202 sec, the radar altitude was valid at 328 sec, and landing took place at about 341 sec past the initiation point. As can be seen from these plots, the complementary filter gave better results for the x-y components during the first 130 sec when TACAN data were used. This is contrary to what was expected because the Kalman filter was designed to remove TACAN bias errors. At the point where MODILS range and azimuth measurements became valid (130 sec), both filters produced excursions in the $y, \dot{y}$ residual components. During this transient period, the complementary filter, with higher fixed gains, had smaller residuals in magnitude. After the transient period, Fig. 22b indicates that the x, $\dot{x}$, y, $\dot{y}$ residuals from the complementary filter had a higher frequency noise content with MODILS data (after 130 sec) than when TACAN data were used. This is also contrary to what was expected, because MODILS data should be less noisy than TACAN data. The velocity component residuals ($\dot{x}$ and $\dot{y}$), based on MODILS data, varied more slowly and were much smoother with the Kalman filter. Thus, the Kalman filter velocity data are preferable from the pilot usage point-of-view in that the displays and guidance commands are smooth. The two filters produced essentially equivalent results for the $z, \dot{z}$ residuals up to a point just before touchdown where the Kalman fil-

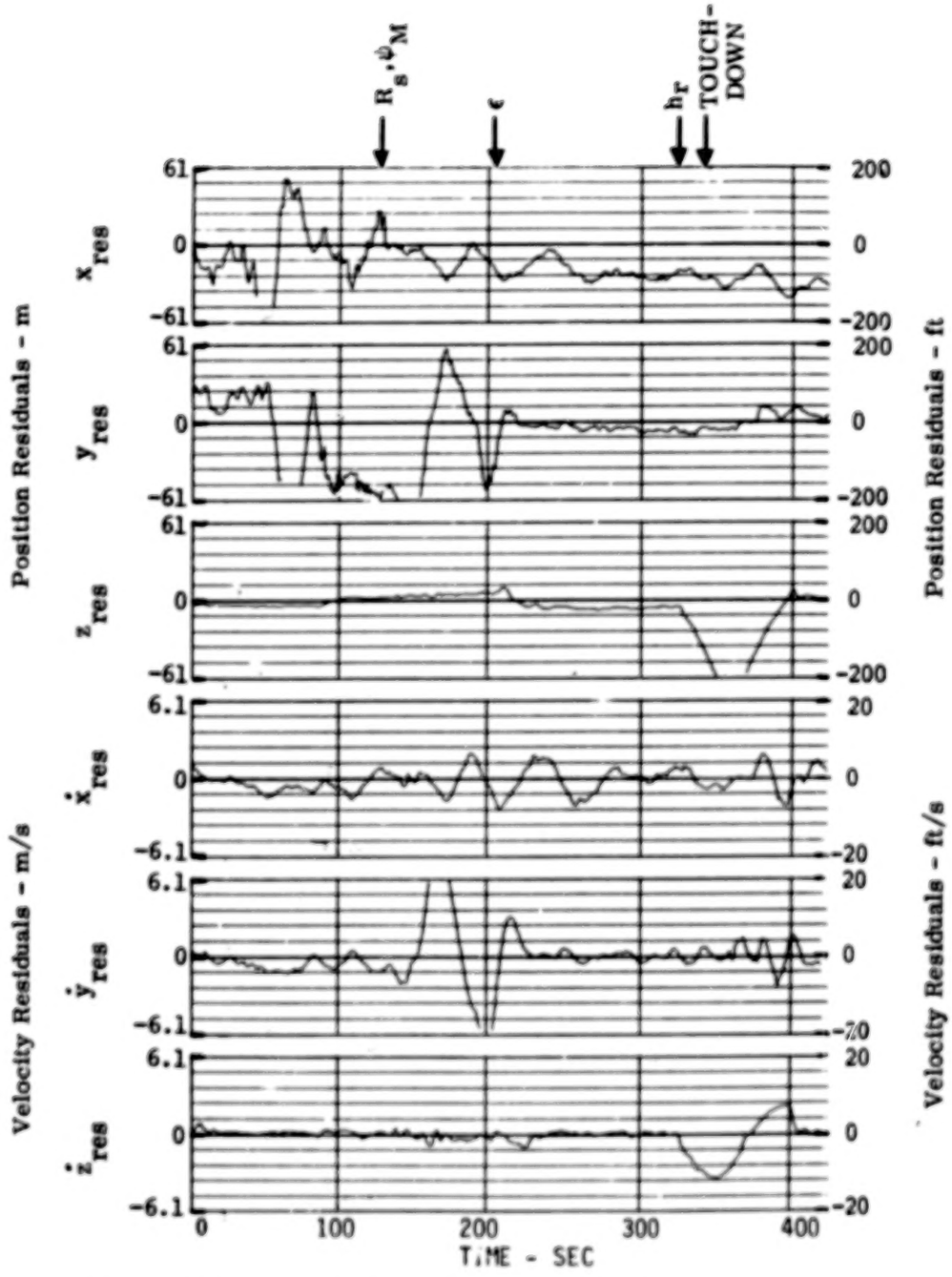


FIGURE 22a.- KALMAN FILTER ERRORS FOR FIRST FLIGHT SEGMENT.

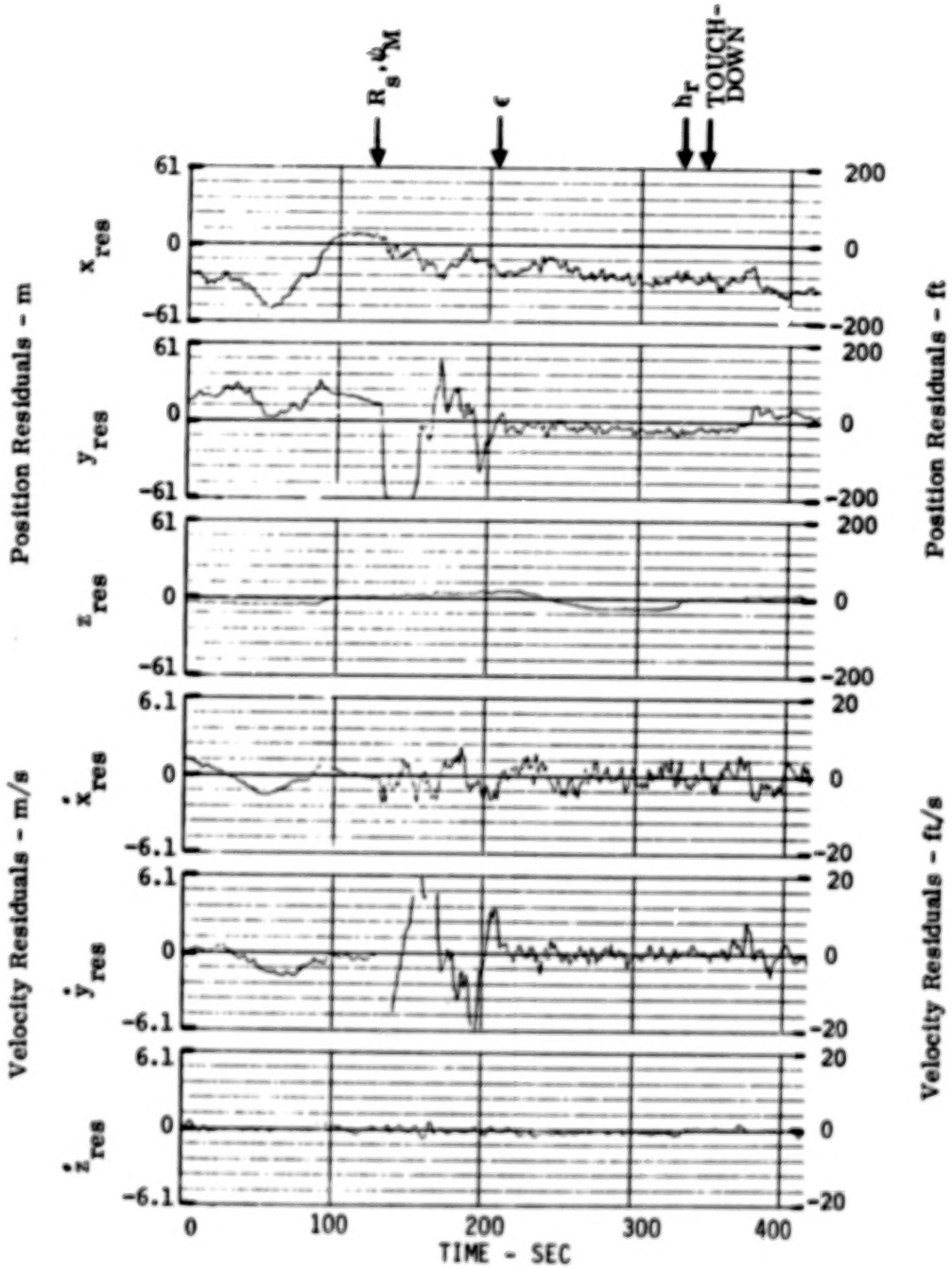


FIGURE 22b.- COMPLEMENTARY FILTER ERRORS FOR FIRST FLIGHT SEGMENT.

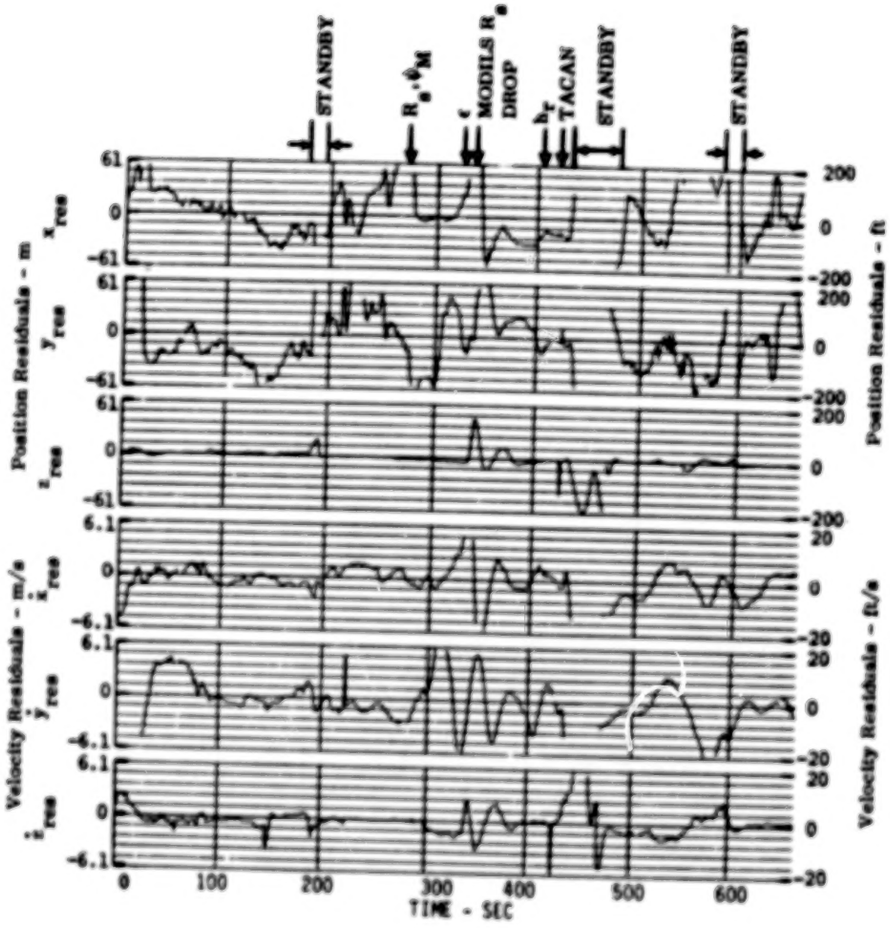


FIGURE 23a.- KALMAN FILTER ERRORS FOR SECOND FLIGHT SEGMENT.

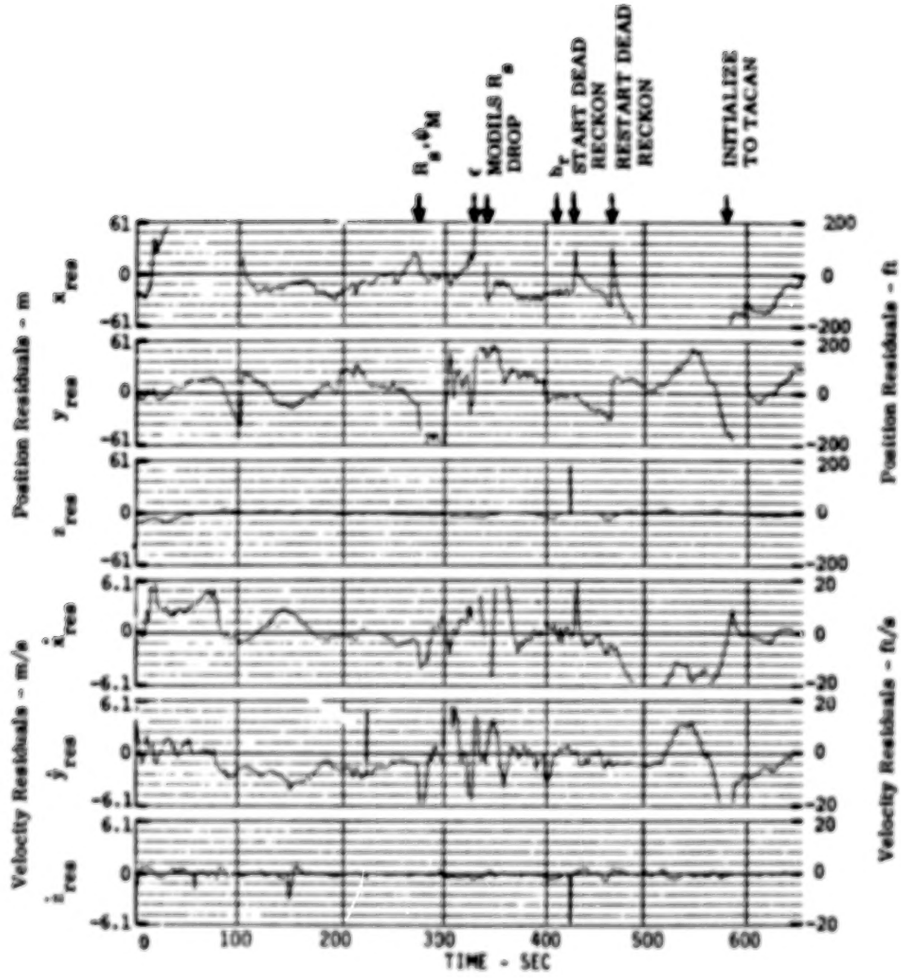


FIGURE 23b.- COMPLEMENTARY FILTER ERRORS FOR SECOND FLIGHT SEGMENT.

This page left intentionally blank.

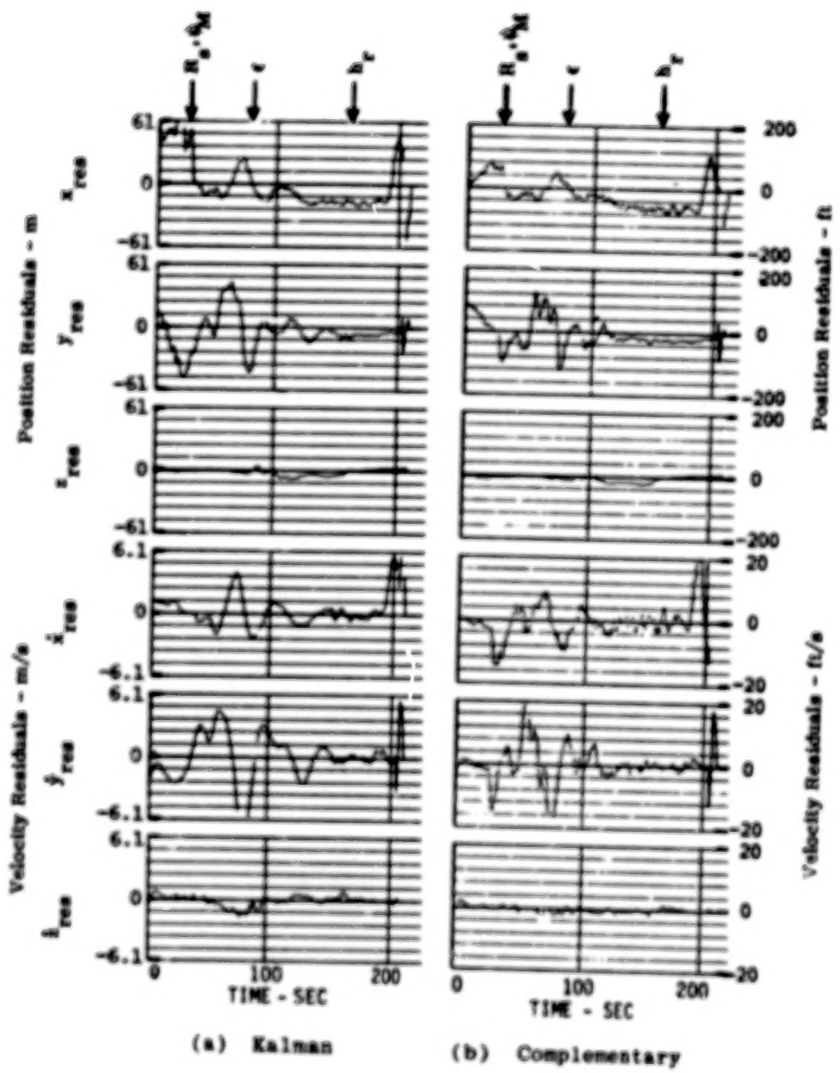


FIGURE 24.- FILTER ERRORS FOR THIRD SEGMENT.

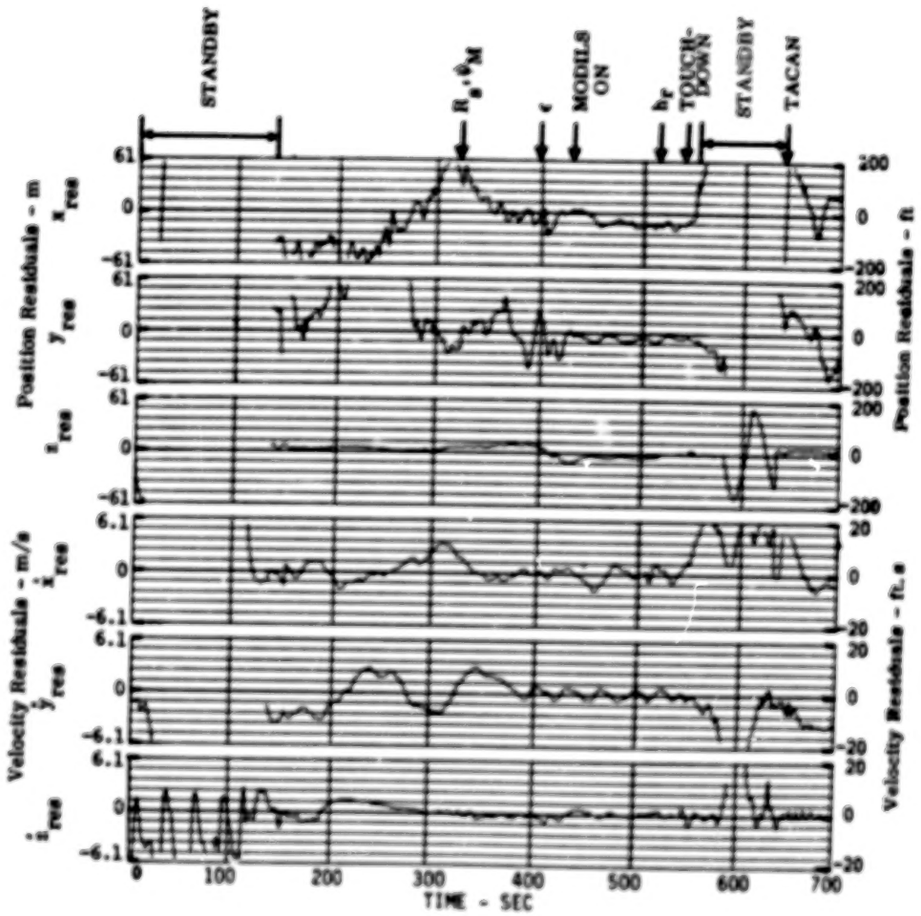


FIGURE 25a.- KALMAN FILTER ERRORS FOR FOURTH FLIGHT SEGMENT.

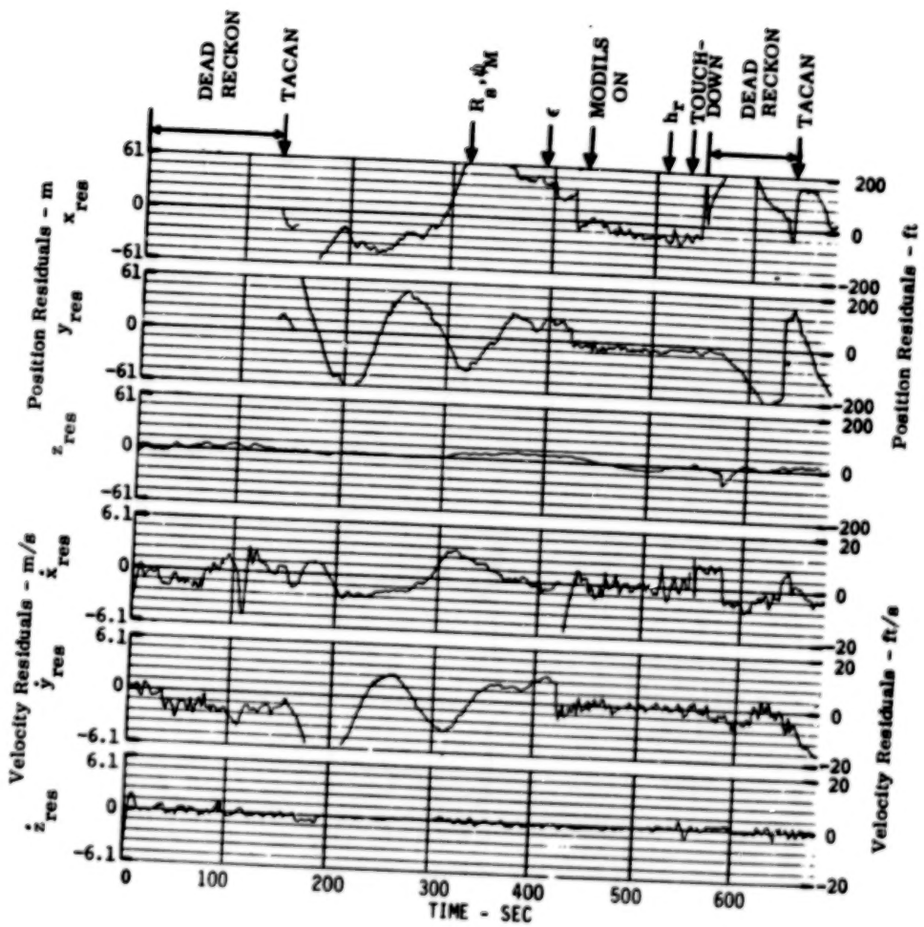
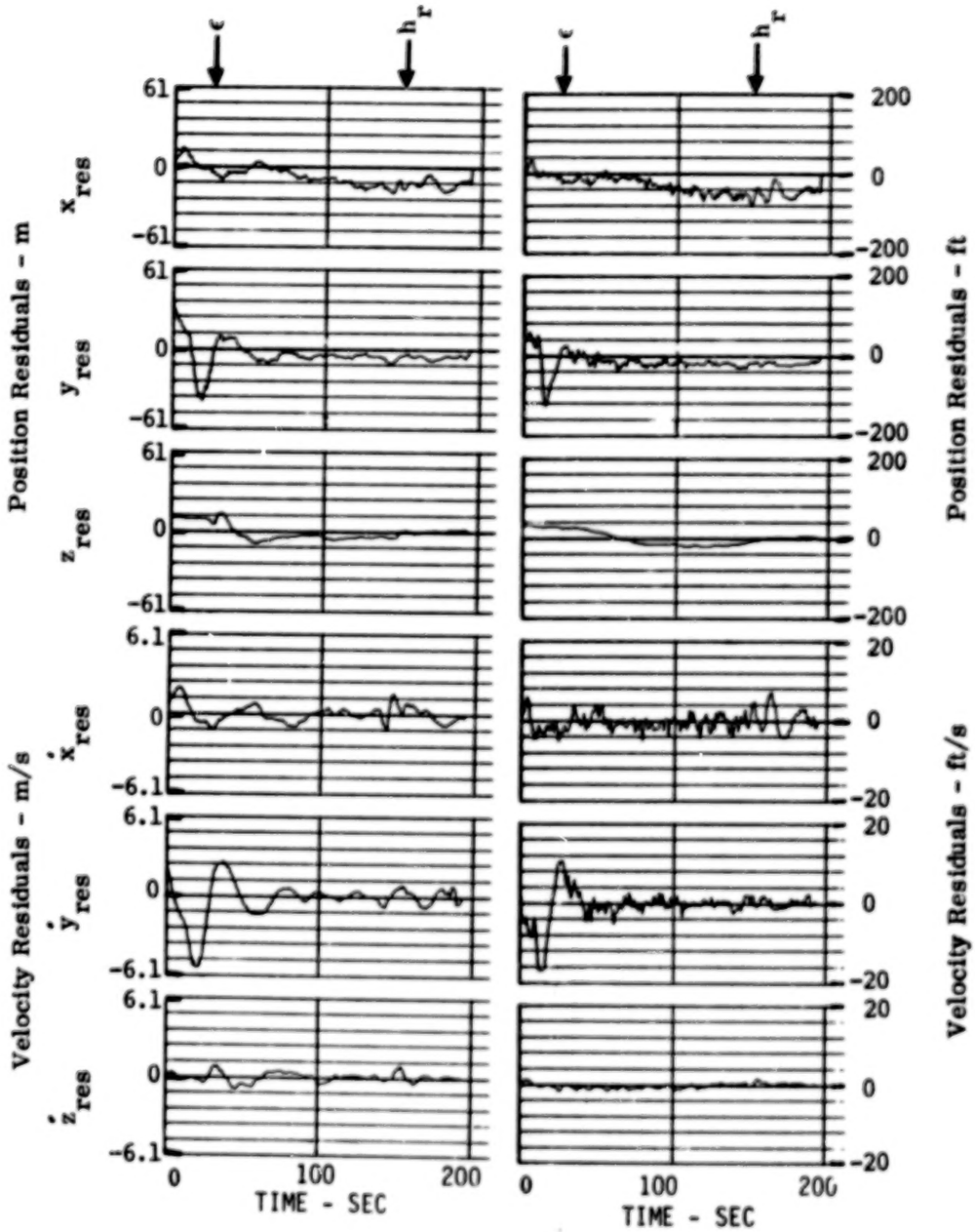


FIGURE 25b.- COMPLEMENTARY FILTER ERRORS FOR FOURTH FLIGHT SEGMENT.



(a) Kalman (b) Complementary

FIGURE 26.- FILTER ERRORS FOR FIFTH SEGMENT.

ter estimates deviated for about 30 sec before reconverging. The cause of the large vertical error was never completely isolated. Except for this particular anomaly in the vertical channel, it can be seen that the two filters produced about the same overall levels of accuracy for the first flight segment.

Figures 23a and 23b show the residuals for the second flight segment which began by using TACAN data. Many different interrelated events occurred during this flight segment, so they are first explained before the filter results are discussed. In this second segment, the MODILS range and azimuth data became valid at 275 sec, the MODILS elevation data became valid at 331 sec, the radar altitude data were valid at 411 sec, and TACAN data were valid to be used again beginning at 421 sec. The MODILS range measurements (R_s) became invalid (drop out) temporarily for about 4 sec beginning at 333 sec.

As noted in Fig. 23a, the Kalman filter was placed in the standby mode for about 10 sec beginning at 185 sec past the initial point. This was done by the pilot to reinitialize the Kalman filter prior to beginning the downwind leg of this segment. The pilot placed the Kalman filter in the standby mode again beginning at 440 sec (after flyover of the MODILS ground equipment). This second standby period lasted about 40 sec, and it was entered because of large filter error buildup due to loss of valid MODILS data and dead-reckoning by the complementary filter. That is, the Kalman filter software was designed to only function when the complementary filter was functioning in its normal mode. At the end of the complementary filter dead-reckoning period (beginning at 585 sec), the pilot again placed the Kalman filter in the standby mode to reinitialize with the valid TACAN data. During the standby modes, the Kalman filter output was held constant to the values it had as it began these periods. After the third standby period was completed (at 600 sec), the Kalman filter began to function again by using valid TACAN data.

After the MODILS data were lost at the runway crossover point (about 440 sec), the complementary filter began a dead-reckoning mode that normally lasts for a period of 120 sec. However, at about 40 sec into this period, it obtained some valid MODILS data. This caused the dead reckoning mode to restart. The dead reckoning mode then continued for 120 sec. At the end of this period, the complementary filter was reinitialized with the valid TACAN data. This is depicted in Fig. 23b.

Because the airborne version of the Kalman filter which was tested did not function during the standby mode, the two filter residuals can only be adequately compared during other periods. This is now done by taking each channel (x,y,z) one at a time from Figs. 23a and 23b.

For the $x,\dot{x}$ components of the second flight segment, up to the first standby period, the Kalman filter had smaller residuals than the complementary filter. After the first standby period and up to when MODILS range and azimuth were accepted, the Kalman filter position residuals demonstrated transient characteristics in recovery from the standby condition. With MODILS data, the Kalman filter position residual converged to a value equivalent to that of the complementary filter. Both filters exhibited similar recovery characteristics after the MODILS range dropout period. The Kalman filter velocity residual was smoother than that of the complementary filter up to the second standby period.

For some reason, the Kalman filter was not initially converged for the $y,\dot{y}$ components during the second segment. Convergence occurred after about 20 sec (Fig. 23a). After this convergence, and before the first standby, the complementary filter had a smaller residual than the Kalman filter. Throughout the rest of this segment, the Kalman filter y channel was mostly in a transient condition reacting to the standby modes and data dropout. It is not clear during this remaining period which filter produced the smaller position residual. Again, the Kalman filter velocity component was smoother than that of the complementary filter.

The $z,\dot{z}$ residuals shown in Fig. 23b for the complementary filter were very small. The Kalman filter position residuals shown in Fig. 23a grew during the MODILS range data dropout. Also, the $\dot{z}$ residual was larger up to the point of MODILS range dropout. The reason the Kalman filter z and $\dot{z}$ errors grew appreciably during the MODILS range dropout is that the filter was using MODILS elevation data for altitude. This involved using estimated values of x and y to compute range in the calculation of the altitude (see Eq. (A.60)). Errors in x and y, therefore, induced errors in the altitude. The complementary filter does a similar calculation; however, this filter uses a blending algorithm [2] to go from baro-altitude data to altitude derived from MODILS elevation data. At the time of the range dropout, the blending had just started; thus, the altitude derived from MODILS elevation data had a negligible effect on the filter. The reason for the larger $\dot{z}$ errors in the Kalman filter, prior to the dropout, is not known.

Figures 24a and 24b show the position and velocity estimate residuals for the third flight segment. The initial (first 100 sec) transient x-y (position) residuals for the Kalman filter had a larger magnitude than those of the complementary filter. The $\dot{x}$-$\dot{y}$ (velocity) residuals for the Kalman filter were smoother than those of the complementary filter. The two filters' $z,\dot{z}$ residuals were both quite small. These results are similar to those of the first flight segment.

Figures 25a and 25b show the position and velocity estimate residuals for the fourth flight segment, which is similar to the second segment. The complementary filter was initially in a dead-reckoning mode, and this lasted for about 140 sec. Also, after landing, the complementary filter again entered a dead-reckoning mode which lasted about 100 sec. During these dead reckoning periods, the Kalman filter was not functioning. Thus, the two filters should only be compared between the two standby/dead-reckon periods. Also, during the fourth flight segment, the MODILS data were not used until about 420 sec, even though these data were valid before this point. (The navaid selection was not automatic; it was selected manually by the pilot.) Thus, up to this point, both filters were using TACAN data.

During the fourth segment shown in Figs. 25a and 25b, while using TACAN data, the Kalman filter $\dot{x}$ position residual was smaller. During MODILS data usage, the Kalman filter $\dot{x}$ velocity residual was smoother. For the $y,\dot{y}$ components the complementary position residual was smaller during TACAN usage. Again, the Kalman filter velocity residual was smoother while using MODILS data. The z position residuals for the two filters were both quite small. The $\dot{z}$ residual for the Kalman filter was initially somewhat larger than that of the complementary filter. These observations are similar to those made from the second flight segment.

Figures 26a and 26b show the position and velocity estimate residuals for the fifth flight segment. For practical purposes, the performances (error magnitudes) of the two filters for this segment of data were essentially equivalent. Again, it can be seen that the velocity residuals for the Kalman filter were much smoother than those of the complementary filter.

Based on the above results, the following conclusions can be made:

(1) The Kalman filter can be implemented and used for navigation purposes during the approach and landing phases of flight.

(2) The Kalman filter, with its three extra state variables, did not provide any marked improvement in accuracy over that provided by the complementary filter. Thus, the TACAN data for these flights can be judged to be fairly accurate. That is, the performance of the complementary filter was not degraded due to the small TACAN range and bearing biases.

(3) The Kalman filter provided smoother velocity residuals than those obtained from the complementary filter. The complementary filter velocity results could potentially be improved by changing the gains indicated in Fig. 8.

Pilot Comments

The mechanization of the Kalman filter in the STOLAND system flight computer was such that both Kalman and complementary filter outputs were simultaneously available. The pilot could select which navigation system was in use by means of the STOLAND keyboard (data entry panel). The pilot flew the system in the fully automatic reference flight path mode for the flight tests so that he was able to observe and compare the two navigation systems without actively flying the aircraft.

The general comments of the test pilot during both preflight simulation tests and the actual flight tests were that the Kalman filter was much better than the complementary filter when both systems were using the MODILS navaids. On both flight and simulation tests, the pilot would alternate between the two systems, and he repeatingly was impressed by the smoothness of the flight and the displays when the Kalman filter was engaged. The pilot could not assess the accuracy of either system. He did report, however, during the final approach on the fifth flight segment where the Kalman filter was engaged, that this was one of the best localizer tracking performances that he had observed. A copy of the pilot flight report is presented as Table 3.

TABLE 3

NASA - AMES RESEARCH CENTER

FLIGHT REPORT

MODEL NO	NO.	DATE OF TEST
DFC-6	STOL-1	12/10/76
JOB ORDER	PROJ. ENG.	PILOT
T-6110	D. Smith	Hardy/Hindson
FLIGHT NO	T.O. GROSS WT/C.G.	FLIGHT TIME
TOE5	≈11,300#	≈1:30

PURPOSE

Collect data on the Kalman filter and make a pilot evaluation of the relative performance between the complementary and Kalman filters.

The A/C was ferried directly to NRC. One 4D RFP3 approach with the complementary filter, 2 4D RFP3 approaches with the Kalman filter, and 2 3D RFP3 approaches switching between filters for pilot evaluation were made. The air was smooth and the surface winds were light to 5 knots. Some detailed comments:

- The HADI went blank on the ferry portion of the flight and came back up on the first touch and go.
- We lost NAV when attempting to switch between NUQ and NRC TAC's on the ferry flight. We had good signals.
- The first approach with the complementary filter was representative of previous flights, i.e., good performance on TAC with some S turning on short final on the MLS.
- The 2nd and 3rd approaches with the Kalman filter and 4D flew the path but never slowed below 120 kts. Difficult to assess the performance from the pilots viewpoint.
- The 4th and 5th approaches alternating between filters and using 3D were completed successfully. There appeared to be little difference (from the pilots viewpoint) while on the TAC portion of the flight but a significant improvement in the MLS performance. The GS track portion of the last approach was flown on the Kalman filter with the only pilot input being an increase in prop RPM at about 1000 feet. The performance was outstanding with the vehicle staying almost exactly in the center of the ILS box until manual takeover at 50 feet. The flight path acceleration bar is also much more steady with the Kalman filter (as in the simulator).
- Had some pitch up and pitch down problems after reengaging the system after the go-arounds.

In summary from the pilots viewpoint the Kalman filter looks very promising. I recommend we evaluate it in the presence of turbulence.

BLANK PAGE

V

POST FLIGHT ANALYSIS

After completing the flight test analysis, some further investigations were conducted by using the NASA Ames CDC 7600 computer. The Kalman filter was coded on the CDC 7600, and then driven with the collected flight test data. The primary goals of the post-flight analysis were to explain the anomalies seen in the Kalman filter flight test results and to develop improvements to the Kalman filter.

The residuals that were produced by the CDC 7600 computer did not duplicate exactly those obtained from the airborne computer (1819A). This is because of differences in the computer characteristics, software, filter initial conditions, and the presence of data dropouts. However, the results were close enough to see the same error characteristics for each segment of flight as those presented in Chapter IV.

Initial focus was placed on correcting the anomalies that appeared in the second flight segment shown in Fig. 23a. The first change that was made was to have the Kalman filter continue to function through all the data. No standby modes were entered in the CDC 7600 program. Thus, the gaps that appeared in the flight results for the x,y residuals (Fig. 23a) were closed in the CDC 7600 results. Figure 27a shows the CDC 7600 simulation results of the Kalman filter position and velocity residuals for the second flight segment. These results are comparable to those from the airborne case shown in Fig. 23a.

The MODILS range data dropout at 330 sec had a more severe effect in the CDC 7600 results. As can be seen in comparing Figs. 23a and 27a, the z position residuals are worse from the CDC 7600 results. Figure 27b shows the TACAN range and bearing bias estimates ($\hat{b}_r$,$\hat{b}_\psi$), baro-altimeter bias estimate ($\hat{b}_h$), and wind estimates ($\hat{w}_x$,$\hat{w}_y$) obtained from the CDC 7600 simulation for this same flight segment. Note that the baro-altitude bias estimates match the character of the residual of the z position estimate shown in Fig. 27a. Thus, the anomalous vertical channel behavior during the second flight segment can be attributed to mismodeling of the various error sources in the vertical channel. Upon reversion to use of baro-altitude data after MODILS dropout, the filter would develop a large altitude bias estimate which did not exist.

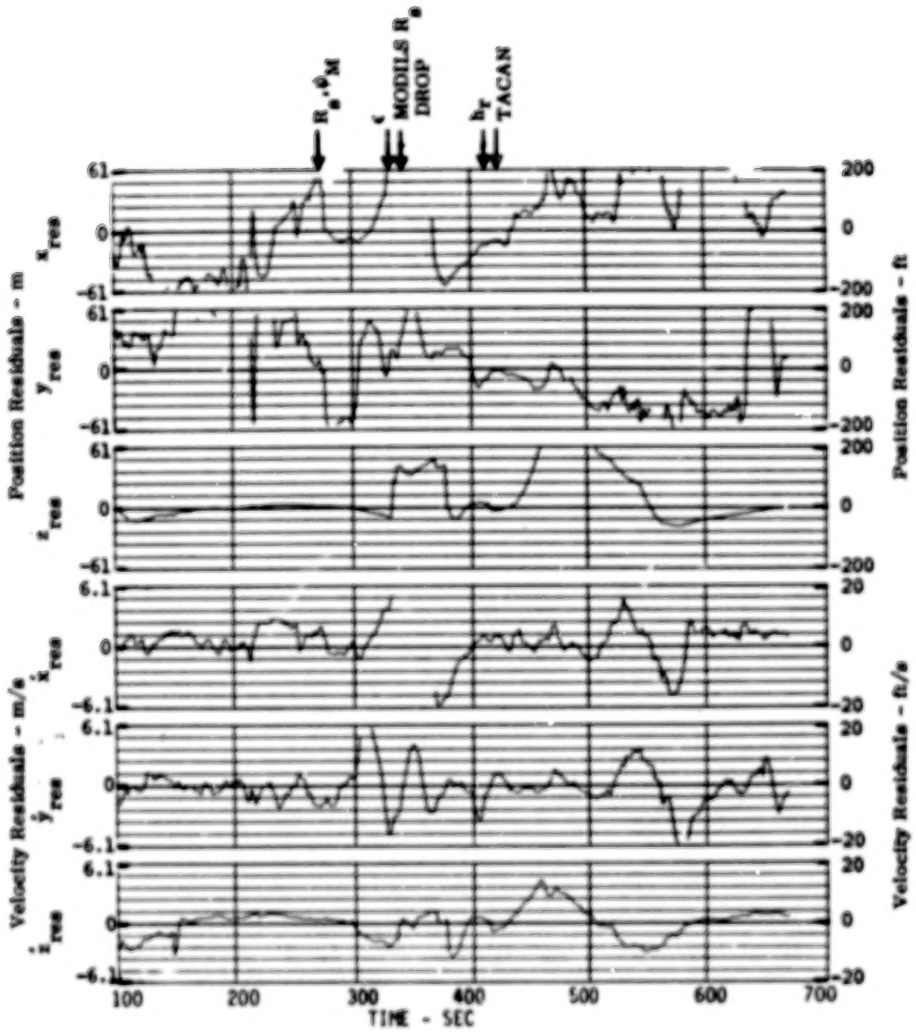


FIGURE 27a.- SECOND FLIGHT SEGMENT KALMAN FILTER RESIDUALS COMPUTED ON CDC 7600.

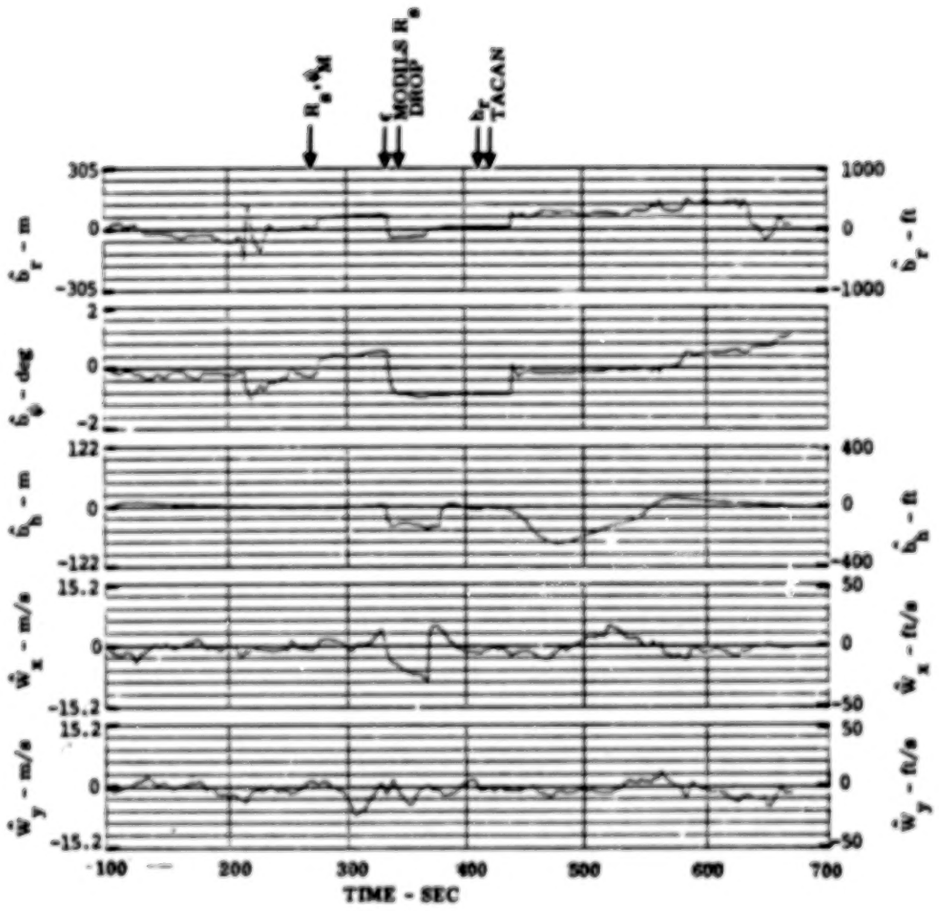


FIGURE 27b.- BIAS AND WIND ESTIMATES FOR SECOND FLIGHT SEGMENT ON CDC 7600.

Further Kalman filter improvements that appeared desirable included:

(a) reduce the velocity errors when MODILS data were in use;

(b) reduce the deleterious effects of the MODILS range dropout; and

(c) explain and correct the anomalous vertical channel behavior which occurred during climbout.

Each of these desired improvements is now discussed.

It was noted in the previous chapter that the velocity residuals appeared to be noisier when using MODILS data than when using TACAN data. It was revealing to compare the MODILS measured aircraft azimuth with that determined by the tracking radars. The azimuth residual ($\psi_{radar} - \psi_{MODILS}$) versus the MODILS azimuth is plotted in Fig. 28 for the five segments recorded. As can be seen from this plot, there is a marked similarity in the azimuth residuals between each segment. This indicates that the MODILS azimuth had a definite angle-dependent error. This is the so-called "ripple effect" which is due to the electronic sweep of the azimuth beam. This may be the source of the above-mentioned velocity errors. It is not known whether this error source is contant or if it fluctuates on a daily basis. If this error is constant, it could be removed by having its error characteristics stored in a tabular form in the airborne computer.

The method used to reduce the effect of the azimuth ripple error was to incorporate a MODILS azimuth data weighting factor in the Kalman filter which is a function of azimuth. That is, the assumed standard deviation of the MODILS azimuth noise error $\sigma_{\psi m}$ used in the filter was given the values indicated in Fig. 29. This function caused the MODILS data to be weighted less during the turning portions of the flights (where the azimuth error due to the ripple effect is continually changing). Consequently, the velocity estimate errors attributed to the ripple would be reduced because the MODILS data would be weighted less during the turn than when on the final approach.

The effects of MODILS range dropout (item (b)) would best be corrected by making a hardware change in the DME receiver. Immediately, on the loss of signal, this change would produce an invalid discrete. The current system uses predicted range for several seconds before marking the data

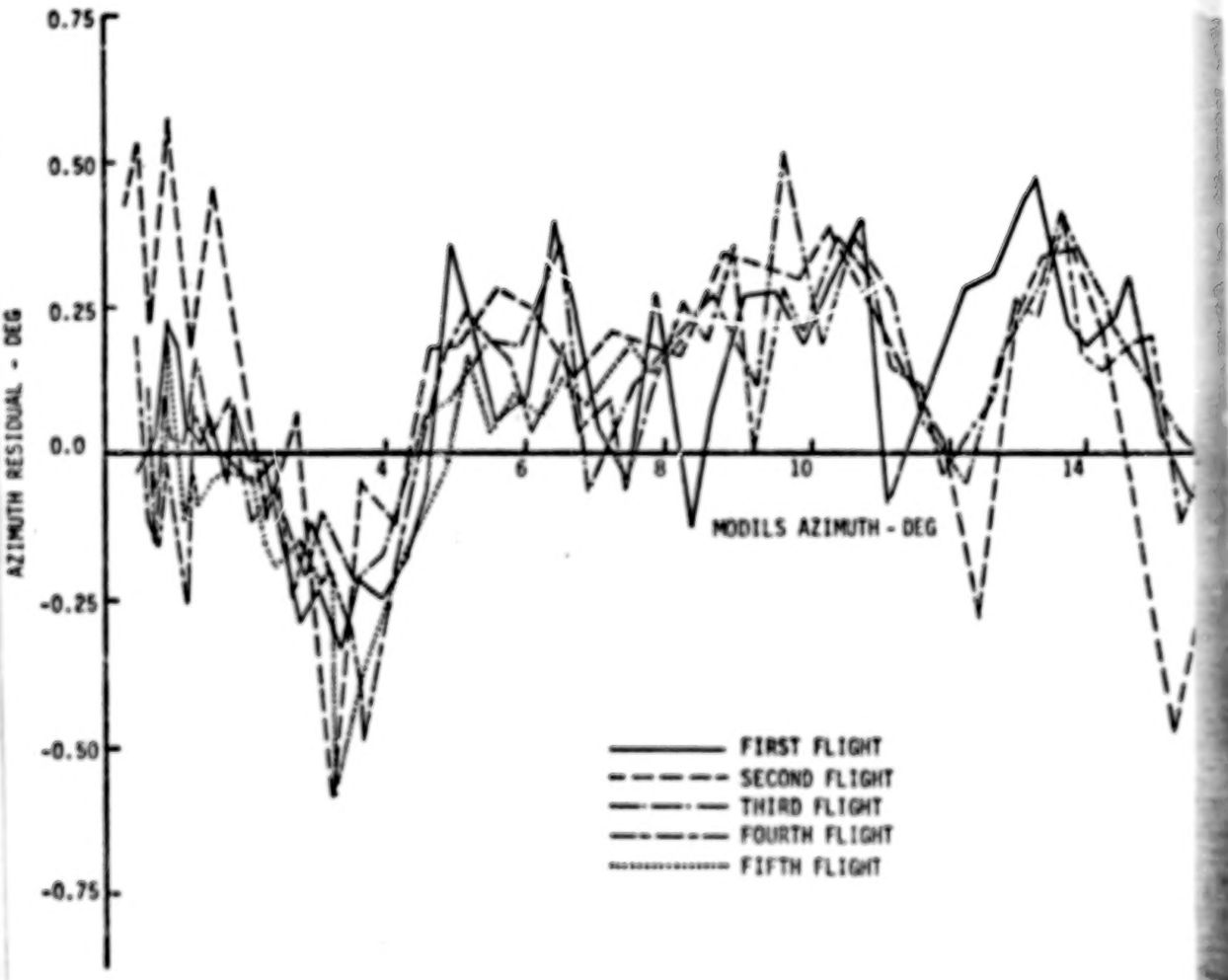
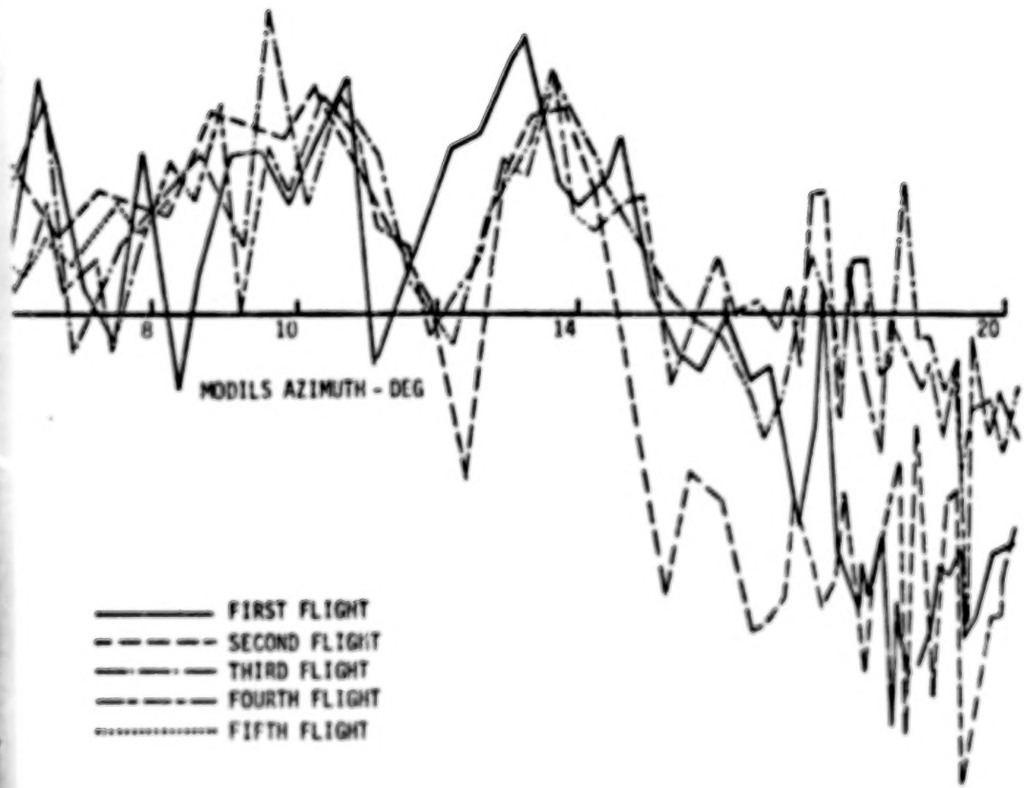


FIGURE 28.-- MODILS AZIMUTH RESIDUALS VERSUS AZIMUTH FOR EACH FLIC



UTH RESIDUALS VERSUS AZIMUTH FOR EACH FLIGHT SEGMENT.

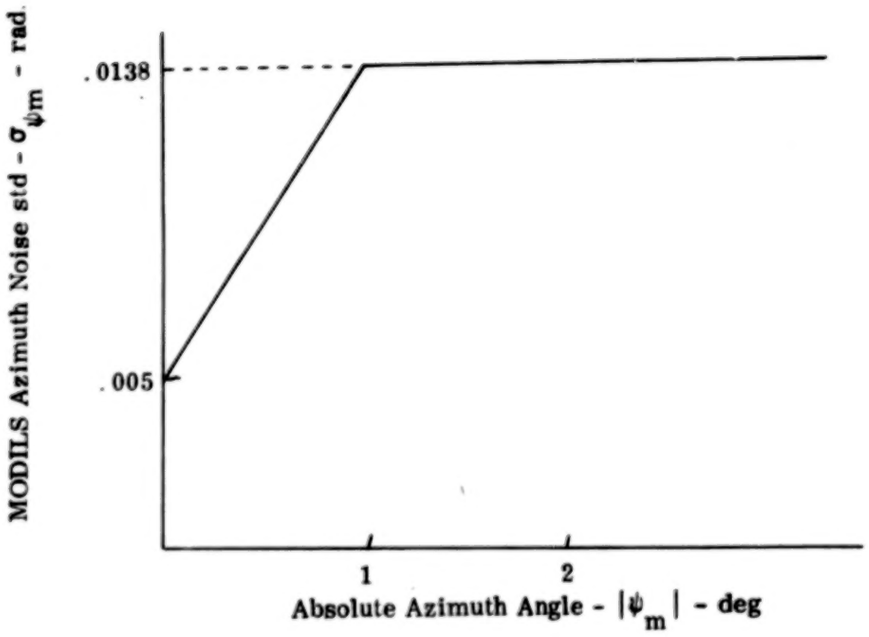


FIGURE 29.- MODILS AZIMUTH NOISE ERROR MODEL USED TO MODIFY KALMAN FILTER RESULTS

invalid. Thus, inaccurate range information continues to be used until the invalid discrete is generated.

To reduce the effect of the MODILS range dropout, a software change was made to the CDC 7600 simulation of the filter to limit the residuals processed by the filter. The logic was arranged so that if the residual was above this threshold (controlled by the estimated variance of the residual), the residual would be ignored (i.e., the measurement would not be processed). Furthermore, the accepted MODILS residual was limited to a second smaller threshold. The limit thresholds chosen for the subsequent tests were:

Range:	300 m (100 ft)
Azimuth:	0.009 rad (0.516 deg)
Derived Altitude:	3 m (10 ft)

This further limited the MODILS data spikes.

To modify the Kalman filter z channel results, the baro-altitude bias was estimated but ignored. That is, the estimated bias had no effect on the residuals used by the filter in the subsequent measurements.

Figure 30a shows the second flight segment residuals recomputed with the above three modifications incorporated into the filter software. Figure 30b shows the corresponding wind and bias estimates. A comparison of Figs. 27a and 30a shows that these corrections make a significant improvement in the overall results. For example, the effect of the range dropout at 330 sec in the vertical channel is practically negligible. The vertical channel transients at 420 sec are gone. Also, the magnitude of the $\dot{x},\dot{y}$ velocity residuals during the turning portion of the flight (before 330 sec) are reduced by up to a factor of two.

By comparing Figs. 27b and 30b, it can be seen that the wind estimates $\hat{w}_x$ and $\hat{w}_y$ are also reduced during the turning period because of the filter modifications. Also, the TACAN bearing bias estimate becomes more nearly constant. It is seen that the estimated TACAN bearing bias and the wind components are quite small. Thus, these terms would have little effect on the navigation system's performance on the day of the flight test.

The above filter modifications were used to reprocess flight data from the first, third, fourth, and fifth segments, in addition to the second segment. Figs. 31a-34a show the residuals from these segments as computed on the CDC 7600 before the filter modifications were made. Figures 31b-34b show the same residuals after the modifications. By comparing these results, it can be seen that the residuals for $z,\dot{z}$ are consistently smaller after the modification. Also, the $\dot{x},\dot{y}$ residuals are consistently reduced during the turning portions of each segment where MODILS azimuth data were used.

Note that in Fig. 31a, the transients in the $z,\dot{z}$ components that exist in Fig. 22a just before touchdown do not exist in the CDC 7600 results. As was mentioned earlier, the cause of the transient shown in Fig. 22a was never isolated.

Another test was made where the baro-altitude bias and the TACAN biases (range and bearing) were removed from the filter. The reason for this change is that TACAN bearing bias is unobservable before the MODILS data are used. This change was tested on the second segment of flight data, and the results are shown in Fig. 35. This figure is compared with the results shown in Fig. 30a. These changes affect the performance produced where TACAN data is used from 100 sec to 280 sec and from 450 sec to the end of the segment. Dropping the TACAN biases appears to decrease the x,y residuals from 100-280 sec. Also, the x residual is improved

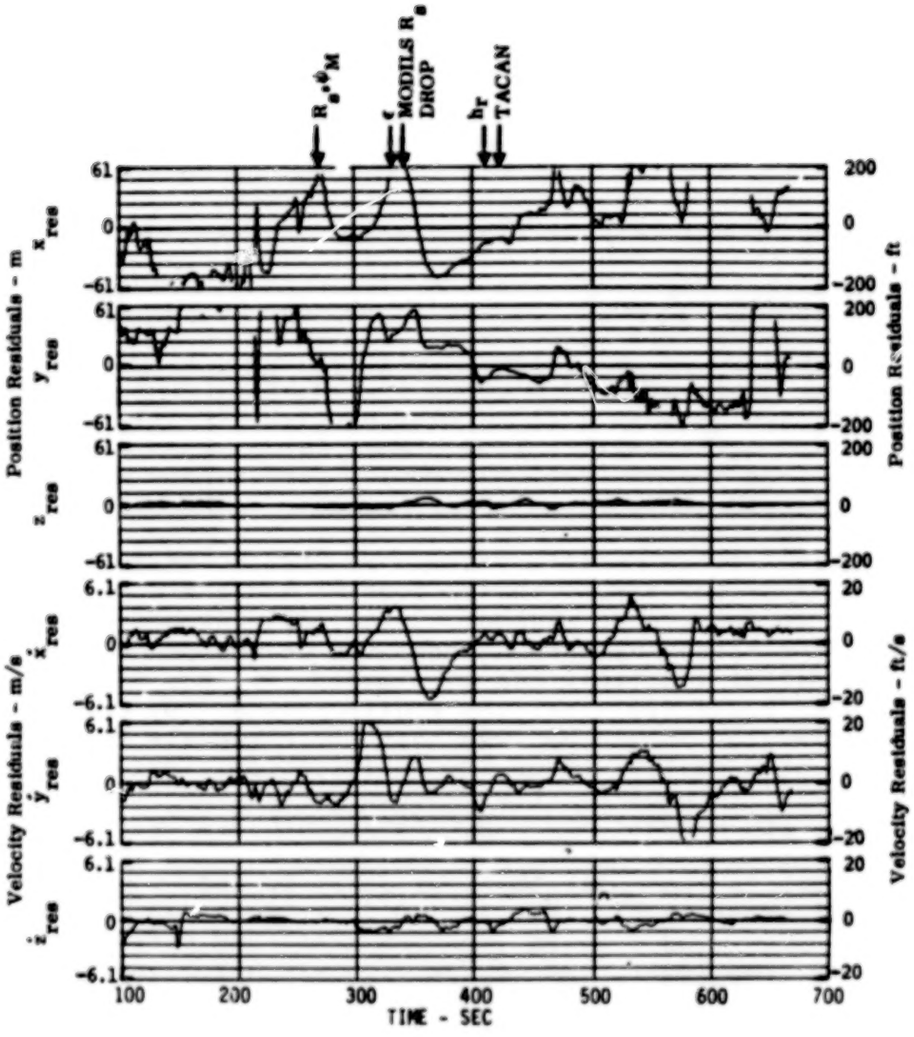


FIGURE 30a.- RECOMPUTED SECOND SEGMENT KALMAN FILTER RESIDUALS WITH MODIFIED FILTER AND BARO-ALTIMETER BIAS REMOVED.

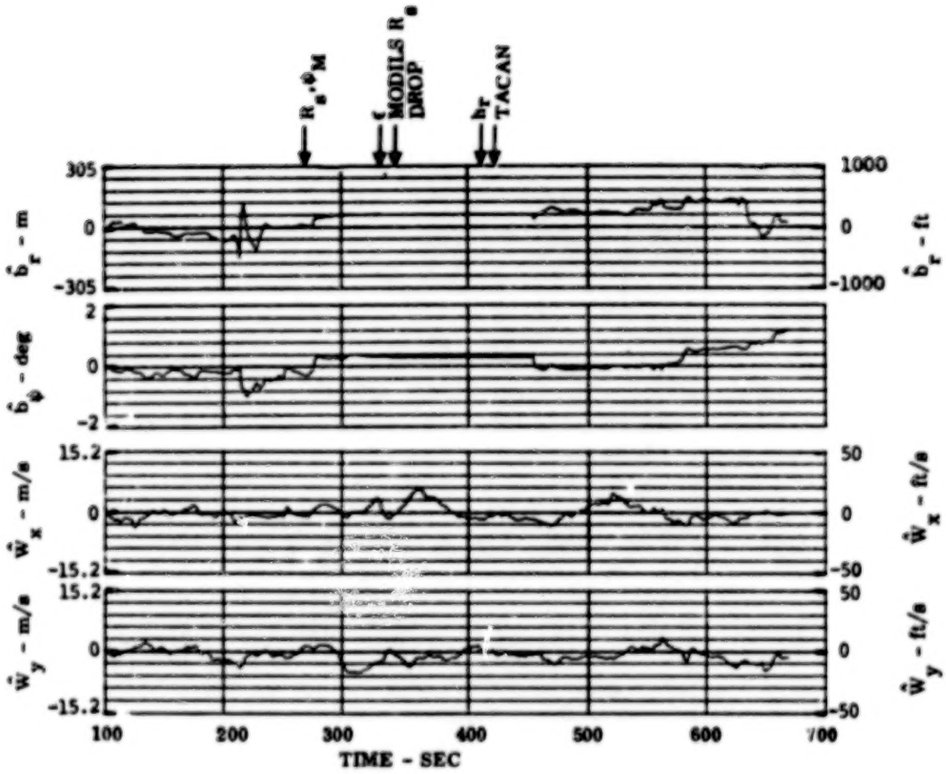


FIGURE 30b.- RECOMPUTED SECOND SEGMENT BIAS AND WIND ESTIMATES WITH MODIFIED FILTER AND BARO-ALTIMETER BIAS REMOVED.

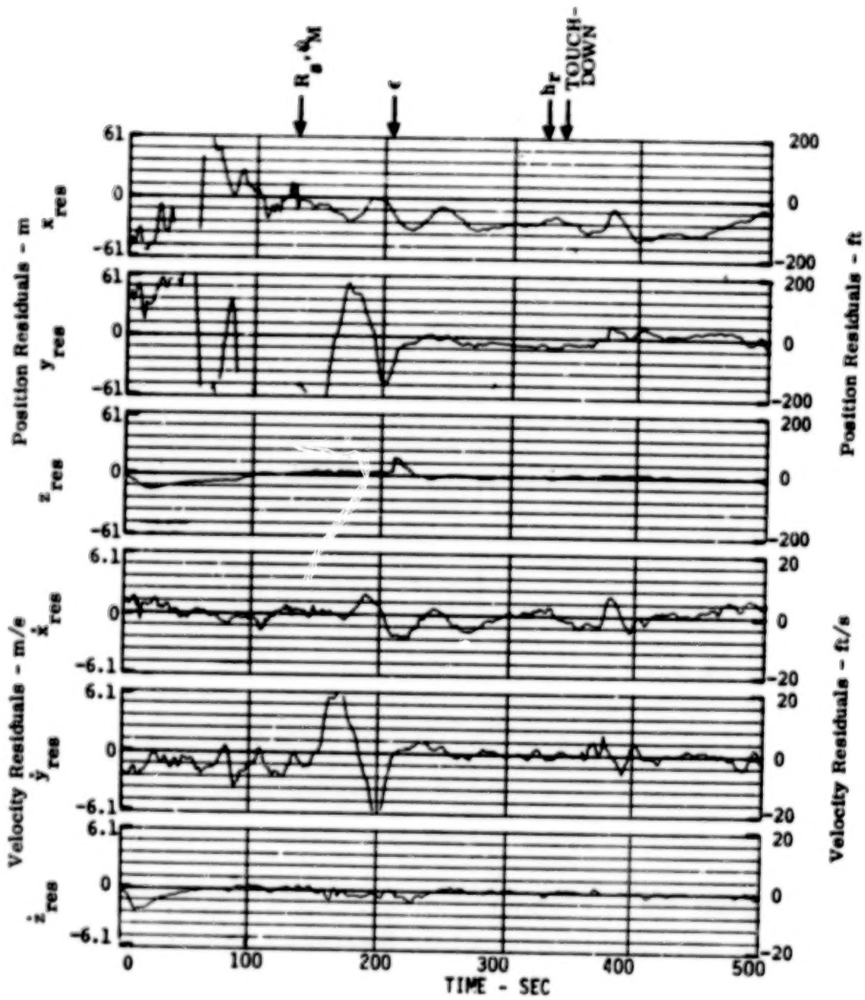


FIGURE 31a.- FIRST FLIGHT SEGMENT KALMAN FILTER RESIDUALS COMPUTED ON CDC 7600 (BEFORE MODIFICATION).

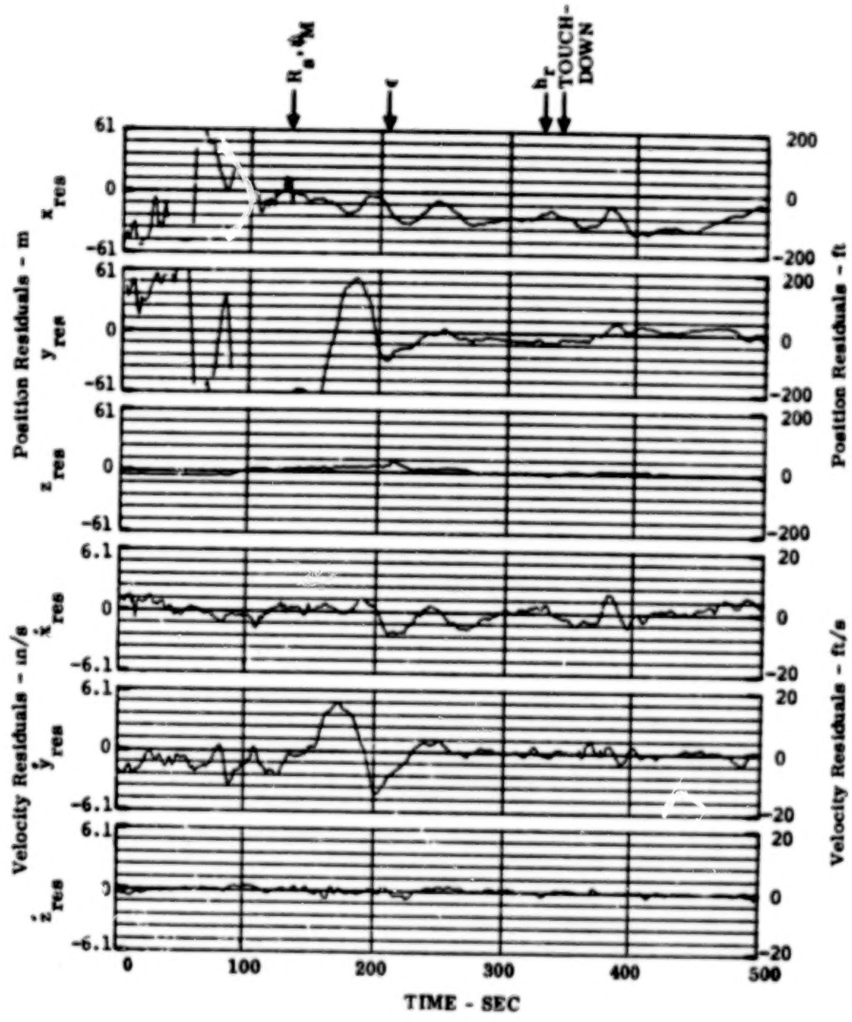


FIGURE 31b.- FIRST FLIGHT SEGMENT (AFTER MODIFICATION).

This page left intentionally blank.

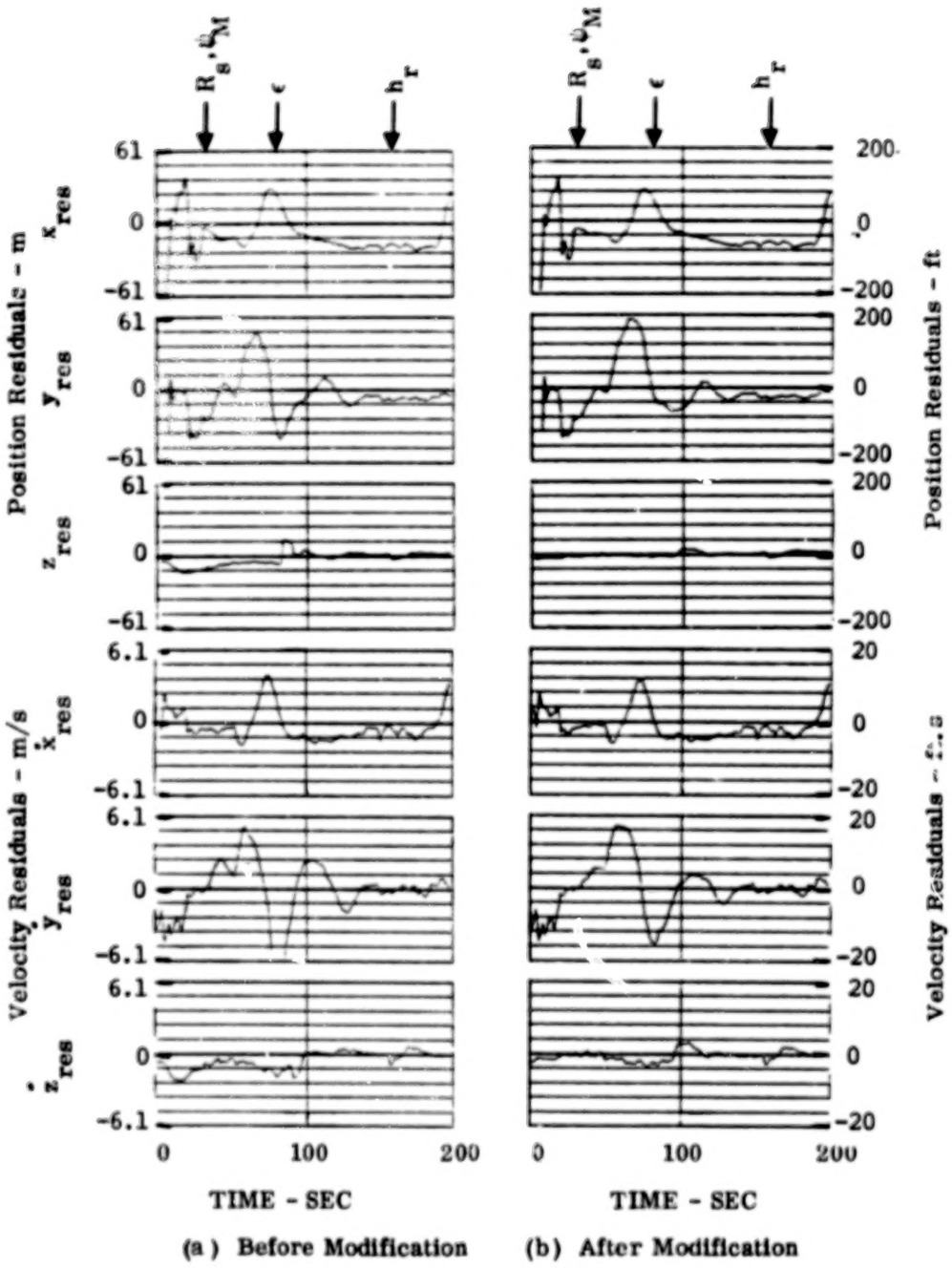


FIGURE 32.- THIRD FLIGHT SEGMENT KALMAN FILTER RESIDUALS COMPUTED ON CDC 7600.

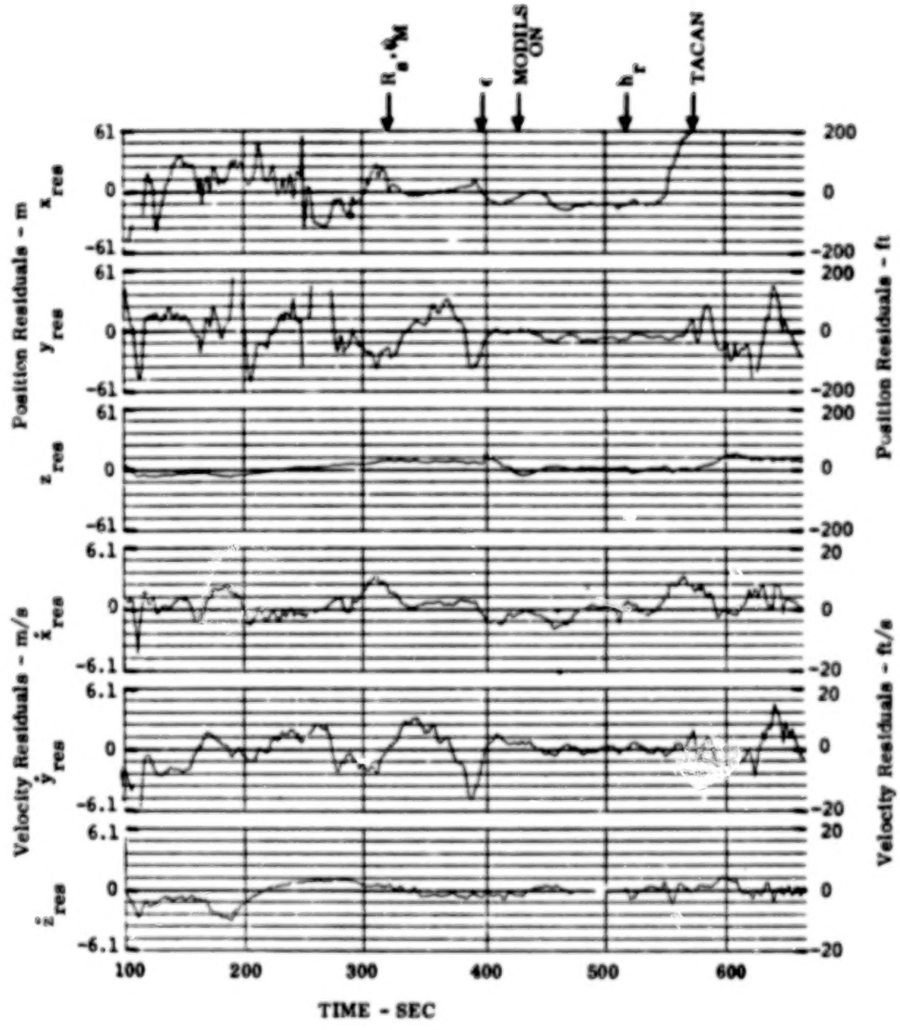


FIGURE 33a.- FOURTH FLIGHT SEGMENT KALMAN FILTER RESIDUALS COMPUTED ON CDC 7600 (BEFORE MODIFICATION).

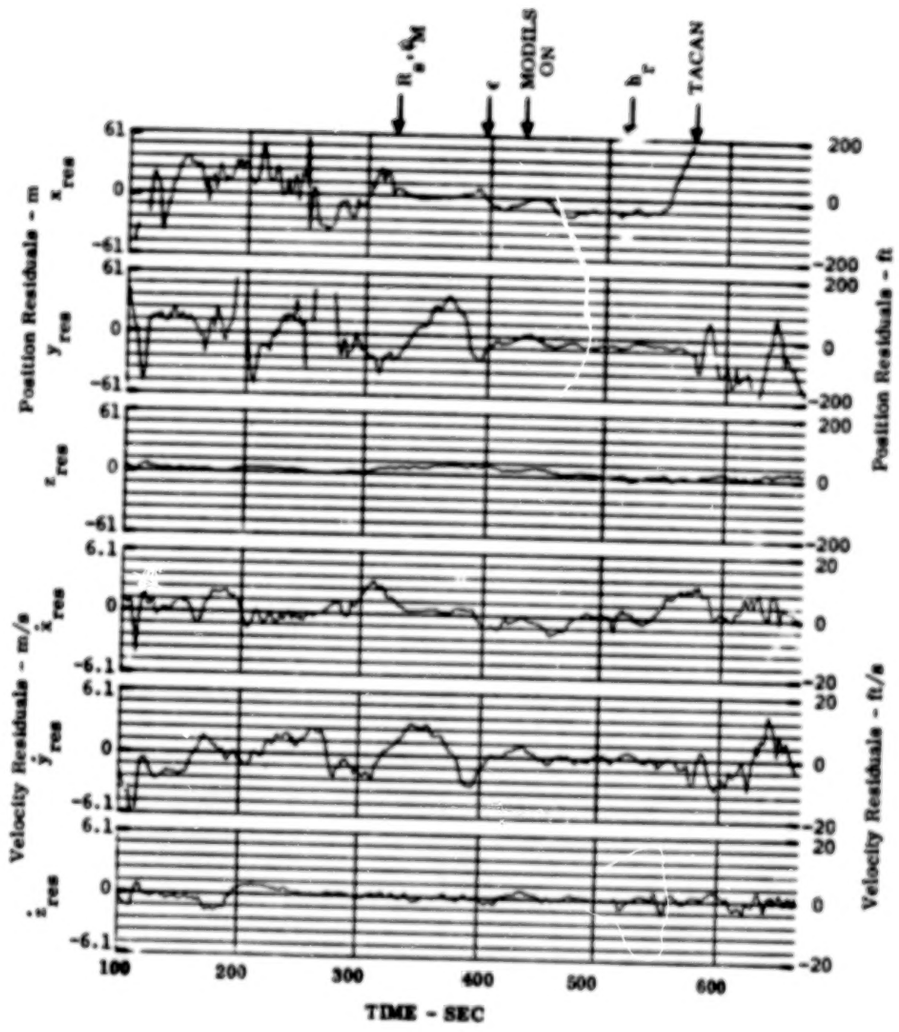


FIGURE 33b.- FOURTH FLIGHT SEGMENT (AFTER MODIFICATION).

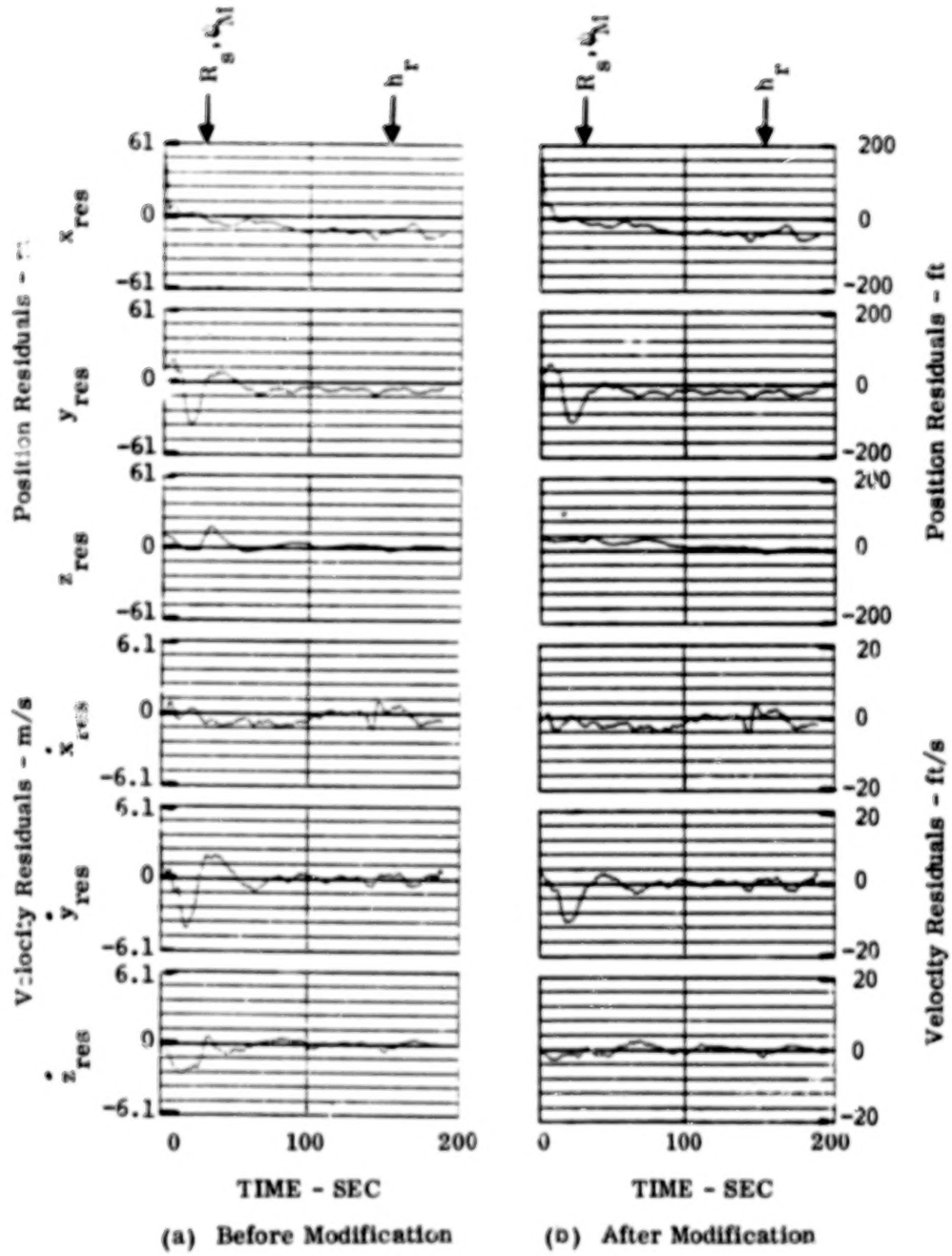


FIGURE 34.- FIFTH FLIGHT SEGMENT KALMAN FILTER RESIDUALS COMPUTED ON CDC 7600.

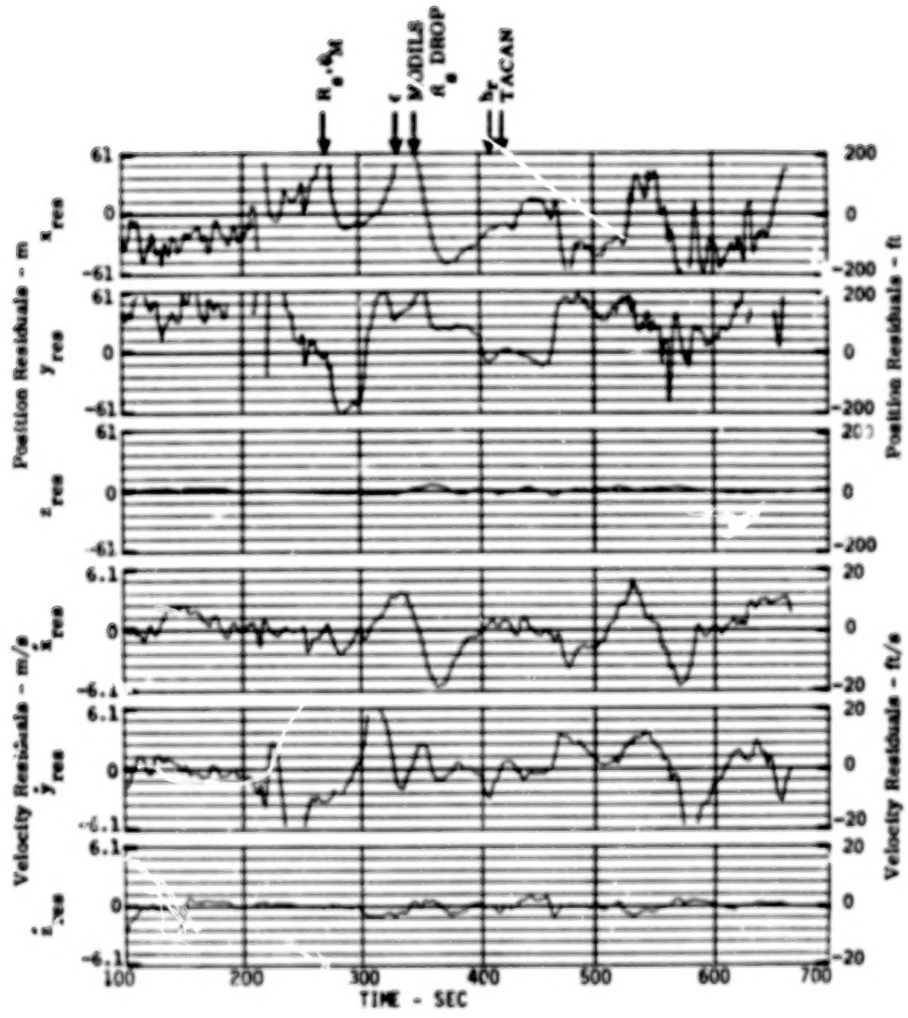


FIGURE 35.- RECOMPUTED SECOND SEGMENT KALMAN FILTER RESIDUALS WITH BARO-ALTIMETER AND TACAN BIASES REMOVED.

from 450 sec on. The y residual appears to be slightly worse than that shown in Fig. 30a. It is concluded that for the data collected during this flight test, the TACAN bias estimates do not provide significant improvement in performance. That is, the TACAN data appeared to be essentially free of bias errors. However, if large TACAN range and bearing biases were actually present, their estimation would have improved the overall results.

In summary, it is concluded that the following software/hardware modifications would significantly improve the Kalman navigation filter tested in the STOLAND system:

(1) Compensate for the ripple effect in the MODILS azimuth signal. The post-flight analysis demonstrated that weighting the azimuth data as a function of the azimuth improved the performance. A better approach would be to make hardware modifications to remove (or reduce) the ripple effect from the MODILS azimuth signal.

(2) Improve the error model for the vertical channel. The post-flight analysis showed that by ignoring the baro-altimeter bias estimate obtained from the Kalman filter, one could remove the anomalous vertical channel errors following revision from radar altimeter data to barometric altimeter data.

(3) Compensate for the effects of dropout in the MODILS DME data. The post-flight analysis showed that by limiting the MODILS residuals, one could reduce the effects of the dropout. A better approach would be to modify the DME receiver so that the hardware flag shows invalid data immediately on loss of signal in the DME receiver.

(4) Change the on board software such that the standby mode does not disengage the Kalman filter. Furthermore, the Kalman filter initialization and operation should not be influenced by the complementary filter modes. The Kalman filter software which was tested was designed so that both the Kalman and complementary filters would use the same navaid information. The Kalman filter operation was controlled to some extent by the complementary filter in an attempt to get good comparative information. The design did not accomplish this purpose; furthermore, it caused serious degradations in the Kalman filter performance in certain regions of the flight.

The above factors (and others) can also enhance the performance of the complementary filter. Reference 13 describes the results of a parallel investigation of complementary filter modifications for achieving improved navigation performance.

TABLE OF CONTENTS

BLANK PAGE

VI

CONCLUSIONS AND RECOMMENDATIONS

This study involved the implementation of an experimental Kalman filter based navigation system in the STOLAND avionics computer (Sperry 1819A). This implementation was then tested aboard the NASA Ames Twin Otter aircraft. The aircraft position and velocity, as derived by the Kalman filter, were measured in the flight test which consisted of five approach, landing, and climbout profiles. The position and velocity were also simultaneously computed by the regular STOLAND navigation software. The results of the Kalman filter and the STOLAND system (complementary filters) derived state variables were then compared. Post-flight simulation studies were also conducted to determine improvements to the Kalman filter configuration.

As a result of these studies, the following conclusions can be made:

(1) It was shown that it is feasible to use a Kalman filter in its full form during the landing phase of flight for navigation computations. The state accuracy provided is equal to or better than that provided by the more conventional complementary filter used in the STOLAND system.

(2) The Kalman filter used in this study provided smoother velocity estimates than that of the complementary filter. This is advantageous for display purposes. The test pilot referred to the Kalman filter performance as "outstanding" in the ability to keep the aircraft on the desired approach beam. The errors in the horizontal position determined by both filters (Kalman and complementary) were about equivalent for the flight profiles flown. The potential improved accuracy of the Kalman filter during usage of TACAN data when large bias errors in range and bearing could be present was not demonstrated in the flight tests because they were conducted on a day when the TACAN biases were small.

(3) The Kalman filter implementation for the tests required about three (3) times the memory and about one and one-half (1.5) times the computer time as that of the complementary filter used

in the comparison. The actual memory requirements for the Kalman filter were about 3000 words, and the time requirements were about 18% of the 1819A computer cycle time of 50 msec. Neither of these requirements is very large when considering modern computer technology.

(4) In future flight tests, improved performance from the Kalman filter could be achieved by eliminating or limiting the barometric altimeter bias error estimate, and by limiting the MODILS derived range, azimuth, and elevation residual magnitudes. Thus, the improved Kalman filter would estimate three components of position, velocity, and acceleration bias, TACAN range and bearing bias, and the two components of the horizontal wind (thirteen state variables). It was practical to keep the vertical channel (z filter) essentially decoupled from the horizontal plane (x-y filter).

The results of this study are being extended at NASA Ames to other aircraft such as the UH-1. They will be useful to future integrated navigation, guidance, and flight control investigations.

Because of the good results from these terminal area studies, it appears possible to use the Kalman filter navigation equations for the entire flight profile consisting of take-off, climbout, cruise, approach, and landing. This would allow using data from multiple VOR/VOR or DME/DME combinations in addition to the VOR/DME (TACAN) used in this study. Then, the Kalman filter should provide greater improvements over the complementary filter. The reference frames necessary for such an implementation may be different, such as those defined in Ref. 14. It is recommended that such a universal implementation of an aided inertial navigation system employing the Kalman filter be developed and flight tested in the future.

APPENDIX A

KALMAN FILTER FORMULATION

This appendix first summarizes the basic Kalman filter equations and the filter design used in the STOLAND Flight Test System. Then, the specific navigation equations used to keep the estimate of the state current and the error equations used to update the filter results in time are summarized. Next, the equations used in the filter to process the external measurements are developed. Finally, measurement preprocessing, interface with the regular STOLAND navigation system, and filter initialization are discussed.

Basic Kalman Filter Principles

The error state dx is defined as the continuous error in the estimate $\hat{X}$ of the aircraft's true state vector X. That is,

$$dx = X - \hat{X} \quad . \tag{A.1}$$

The aircraft navigation error state considered in this study had fourteen variables (elements) which are:

- 3 - element position error vector with respect to the runway,
- 3 - element velocity error vector with respect to the runway,
- 3 - element acceleration bias vector,
- 2 - element error in horizontal wind vector,
- 1 - element TACAN range bias,
- 1 - element TACAN bearing bias,
- 1 - element baro-altimeter bias.

For mechanization convenience, these variables were separated into a group of ten associated with the aircraft horizontal position (x-y filter) and a group of four associated with the aircraft vertical position (z filter). This decoupling is explained more fully later. The following development initially ignores the decoupling.

The fourteen elements of dx are assumed to be small so that the dynamics which describe their time rate of change can be modeled by the linear matrix differential equation,

$$\dot{dx} = F_x dx + F_\eta \eta \quad . \tag{A.2}$$

Here,

dx = the n (14) element error state vector,

F_x = an n x n system dynamics matrix,

F_η = an n x m error distribution matrix,

η = a vector of m random forcing functions for compensation of error growth caused by unmodeled error sources.

The objective of the Kalman filter is to estimate the error state dx so that: (a) the true aircraft state can be more accurately known, and (b) the effects of the various sensor biases and wind can be removed by compensation. The filter estimate of the error state is defined as $\hat{dx}$. The functions of the Kalman filter algorithm are to: (a) carry the error state estimate dx along in time, and (b) to update (or increment) $\hat{dx}$ based on external measurement information. Then, on a regular basis, the error estimate $\hat{dx}$ is used to correct the total state estimate $\hat{X}$.

Because Eq. (A.2) represents a linear system, the error state dx can be advanced from time point to time point by use of the state transition matrix. The approximate solution to Eq. (A.2) for dx at time point t_{k+1}, given $dx(t_k)$, is

$$dx(t_{k+1}) = \Phi(t_{k+1};t_k)\, dx(t_k) + \Phi_u(t_{k+1};t_k)\, u(t_k). \tag{A.3}$$

Here,

Φ = the state transition matrix,

Φ_u = the forcing function sensitivity matrix,

$u(t_k)$ = a constant (in the interval t_k to t_{k+1}) vector for approximating the effects of the random vector, η, of Eq. (A.2).

The Kalman filter utilized in this study was designed to minimize effects caused by: (a) numerical calculation errors such as truncation, and (b) modeling errors resulting from various approximations. Past experience has shown that the square root implementation [10] of the Kalman filter algorithm can reduce the effects of the numerical errors to insignificant levels. The square root implementation was therefore incorporated into the design used in this study. Modeling errors were compensated by the appropriate use of random forcing functions. This technique causes the more recent measurements to be weighted more than past measurements; therefore, the estimate tends to follow the more recent measurements.

An essential part of the Kalman filter is the covariance matrix $P(t_k)$ of the error state dx at each time point t_k. This matrix is given by

$$P(t_k) = W(t_k)\, W^T(t_k) = E\{dx(t_k)dx^T(t_k)\} \quad . \qquad (A.4)$$

where

$W(t_k)$ = the square root of the covariance $P(t_k)$ (W^T is calculated in the square root implementation of the filter).

$E\{\ \}$ = the expected value operator.

It is assumed that $u(t_k)$ of Eq. (A.3) is a random independent vector such that

$$E\{u(t_{k+i})u^T(t_{k+\ell})\} = 0 \quad ; \qquad i \neq \ell$$

$$= U(t_{k+i})U^T(t_{k+i}) \quad . \quad i = \ell \qquad (A.5)$$

It is necessary to update the covariance matrix P from one consecutive time point t_k to the next t_{k+1}. The appropriate use of the expected value operator with Eq. (A.3) gives the time update of the covariance matrix as,

$$P(t_{k+1}) = W(t_{k+1})W^T(t_{k+1}) = [\Phi W(t_k)\Phi_u U] \begin{bmatrix} W^T(t_k)\Phi^T \\ U^T\Phi_u^T \end{bmatrix} . \qquad (A.6)$$

From Eq. (A.6), it is seen that one can form $W^T(t_{k+1})$ as follows:

$$W^T(t_{k+1}) = \begin{bmatrix} W^T(t_k)\Phi^T(t_{k+1};t_k) \\ U^T(t_k)\Phi_u^T(t_{k+1};t_k) \end{bmatrix} . \qquad (A.7)$$

The matrix, $W^T(t_{k+1})$ of Eq. (A.7) has dimension $(n+m) \times n$. The Householder algorithm described in Ref. 10 is used in the filter mechanization to reduce this matrix to an upper triangular form. That is, all the terms below the diagonal are zero in the reduced matrix. The matrix reduction algorithm leaves the product WW^T invariant.

A discrete measurement residual y_m is defined as the difference between the external state measurements $Y_m(X,t)$ of the aircraft (from available navaids, air data sensors, etc.) and the computed value $\hat{Y}(\hat{X},t_m)$ of the measurements based on the continuously available navigation equation state estimate $\hat{X}$. Explicit definitions of the navigation equations and computed measurement equations are given in the next section. It is assumed that the measurement residual is related to the error state dx at the time point t_m when the measurement is made by the equation

$$y_m(t_m) = Y_m(X,t_m) - \hat{Y}(\hat{X},t_m),$$

$$= Hdx(t_m) + q \quad . \qquad (A.8)$$

Here,

H = external measurement distribution (sensitivity) matrix, and

q = the random noise error in the external measurement.

For an individual measurement, H is a row vector. The Kalman filter is based upon the structure of Eqs. (A.2) and (A.8) and the assumed Gaussian statistical properties that describe the vectors η and q.

Similar to Eq. (A.8), the estimated measurement residual at time point t_m is assumed to b

$$\hat{y}_m(t_m) = Hd\hat{x}(t_m) \quad . \tag{A.9}$$

Here,

$\hat{y}_m(t_m)$ = the computed value of the position measurement residual based on the error state estimate $d\hat{x}(t_m)$.

External measurements can be taken rapidly at arbitrary time points t_m. For the terminal area navigation system of this study, it is computationally inefficient to advance the square root covariance W^T of Eq. (A.7) to each arbitrary time point t_m to process each measurement. Thus, the Kalman filter is mechanized to operate with cyclic reference times t_k, t_{k+1}, t_{k+2}, etc. Then all external measurements taken between t_k and t_{k+1} are used to adjust the estimate dx at the time point t_k. For the system mechanized in this study, $(t_{k+1}-t_k)$ was set at 1.5 sec.

The error in the estimated residual $\hat{y}_m(t_m)$ can be found from Eqs. (A.3), (A.8), and (A.9) to be approximately

$$\begin{aligned} \Delta y(t_m) &= y_m(t_m) - \hat{y}_m(t_m) \quad , \\ &= y_m(t_m) - H\Phi(t_m;t_k)\, d\hat{x}(t_k) \quad . \end{aligned} \tag{A.10}$$

This neglects the effects of the driving term η, in Eq. (A.2). Experience has shown that this simplification is justified when the update interval $(t_{k+1}-t_k)$ is small compared to the natural frequencies of error growth in the navigation equations.

To update the estimated error state $d\hat{x}(t_k)$ from each of the measurement residuals $\Delta y(t_m)$, the following computations are typically made. Let

$$H_m = H\Phi(t_m;t_k) \quad ,$$

$$K_m = W(t_k)W^T(t_k)H_m^T/(H_mW(t_k)W^T(t_k)H_m^T + Q) \quad . \qquad (A.11)$$

Here, Q is the assumed variance of the random error q in the measurement,

$$Q = E(q^2) \quad . \qquad (A.12)$$

Then, the new estimate of the error state following inclusion of the measurement would be,

$$d\hat{x}(t_k)_a = d\hat{x}(t_k)_b + K_m\Delta y(t_m) \quad . \qquad (A.13)$$

Here, the subscript notation $(\ )_a$ and $(\ )_b$ means before and after inclusion of the measurement, respectively.

The square root covariance matrix $W^T(t_k)$, after inclusion of the measurement, would be updated by,

$$W^T(t_k)_a = W^T(t_k)_b - BC^T/D \quad , \qquad (A.14)$$

where

$$B = W^T(t_k)_bH_m^T \quad ,$$

$$C = W(t_k)_bB \quad ,$$

$$D = (B^TB + Q)\ [1 + \sqrt{Q/(B^TB + Q)}] \quad .$$

Equation (A.14) is referred to as Potter's algorithm (see Ref. 10).

In order to reduce the number of operations in the airborne computer further, the residuals of each external measurement were accumulated (and effectively averaged) over the 1.5 sec period rather than individually processed. That is, instead of using Eqs. (A.10)-(A.14) everytime a new measurement is taken, the residuals of a particular measurement variable (e.g., TACAN range) are accumulated over the 1.5 sec period according to

$$y(t_m) = Y - \hat{Y}(\hat{X}, t_m) \qquad \text{(A.15a)}$$

$$y_s = y_s + \Delta y(t_m) \qquad \text{(A.15b)}$$

Note that the residual in Eq. (A.15a) is based on the estimated total measurement $\hat{Y}$ from Eq. (A.8). This estimate is based on the estimated total state $\hat{X}$ at time t_m which comes from the navigation equations.

The measurement sensitivity matrix H_m of Eq. (A.11) is computed and accumulated simultaneously with the residual accumulation. That is,

$$H_m(t_m) = H\Phi(t_m; t_k) \ , \qquad \text{(A.16a)}$$

$$H_{ms} = H_{ms} + H_m(t_m) \ , \qquad \text{(A.16b)}$$

are used to accumulate H_m over the 1.5 sec period. Furthermore, the variance Q of Eqs. (A.11) and (A.12) is replaced with the assumed variance Q_s of the random noise error in the accumulated residual. More mechanization details are given on the residual sum processing later.

Now define the variance in the accumulated residual of an individual measurement as

$$S = H_{ms}W(t_k)W^T(t_k)H_{ms}^T + Q_s \ . \qquad \text{(A.17)}$$

Then, after the last external measurement has been taken and y_s, H_{ms}, Q_s and S have been computed during the 1.5 sec period, the estimated error state $d\hat{x}(t_k)$ is updated according to

$$dx(t_k)_a = dx(t_k)_b + WW^TH_{ms}^T[y_s - H_{ms}dx(t_k)_b]/S \ . \qquad \text{(A.18)}$$

As can be seen, Eq. (A.18) is a variation of Eqs. (A.11) and (A.13).

Also, for each accumulated measurement, the square root covariance is updated according to

$$W^T = W^T - W^T H_{ms}^T H_{ms} W W^T / [S(1 + \sqrt{Q_s/S})] \quad , \qquad (A.19)$$

Equation (A.19) is a variation of Eq. (A.14).

Equations (A.18) and (A.19) are repeated for each of the different measurements in effect (e.g., TACAN range and bearing, air data, and baro-altitude). This produces an updated error state estimate $d\hat{x}(t_k)$, at time point t_k. This estimate is then advanced to the time point t_{k+1} according to

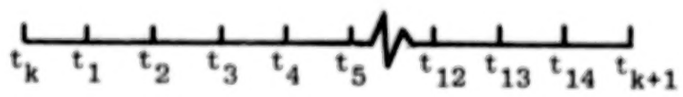
$$d\hat{x}(t_{k+1}) = \Phi(t_{k+1};t_k)\, d\hat{x}(t_k) \quad . \qquad (A.20)$$

The estimated total state is then updated by

$$\hat{X}(t_{k+1})_a = \hat{X}(t_{k+1})_b + d\hat{x}(t_{k+1}) \quad , \qquad (A.21)$$

where $\hat{X}(t_{k+1})_b$ is the value obtained from the navigation equations. After completion of Eqs. (A.21), $d\hat{x}(t_{k+1})$ is set to zero, the square root covariance is advanced to t_{k+1} according to Eqs. (A.7), and the process is repeated for the next cycle.

In summary, the mechanized filter was designed to operate along a time line illustrated in the sketch below:



Each of the time points t_k, t_1, etc. are 0.1 sec apart. The major time interval between points t_k and t_{k+1} is 1.5 sec. At the start of the sequence, the filter has its covariance matrix P referenced to time point t_k. At the time points

$t_k, t_1, \ldots, t_{14}$, external measurements are accepted by the filter, and measurement residuals and partials are computed, accumulated, and saved in the preprocessing routines. After the filter processes the measurements at the time point t_{14}, the residual sums and partials are used for updating the incremental state estimate at t_k. Then, the locations used for the preprocessing are cleared for use in preprocessing the external measurements at t_{k+1} and the subsequent time points.

The residual sums are processed by the filter, and the incremental state change is computed. When these calculations have been completed, the estimated state change is advanced and added to the total state at the time point t_{k+1}. Meanwhile, other computations update the covariance matrix to the time point t_{k+1}, and the logic sets up the filter for processing the residual sums taken during the next major time interval. The on board program operations for executing this logic in the Sperry 1819A computer are described in Appendix B.

Navigation and Error Equations

The three-axis Kalman filter used in the Ames STOLAND Flight Test System is depicted in Fig. 10. The vertical elements of the estimated state (z-filter) discussed in Ref. 6, remain decoupled from the horizontal elements (x-y filter) discussed in Ref. 5. Attitude and heading data and three body-mounted accelerometers are used to transfer the acceleration measurements into the runway reference frame. Existing STOLAND 1819A software performs this transformation at a 20 Hz frequency.

The navigation equations used to keep the state estimate current integrate the terms

$$a_{sx} = \ddot{x}_r + \hat{b}_{ax} \quad ,$$

$$a_{sy} = \ddot{y}_r + \hat{b}_{ay} \quad ,$$

$$a_{sz} = \ddot{z}_r + \hat{b}_{az} \quad , \qquad \text{(A.22)}$$

where

$\ddot{x}_r, \ddot{y}_r, \ddot{z}_r$ = raw acceleration in the runway reference frame as computed by existing STOLAND software.

$\hat{b}_{ax}, \hat{b}_{ay}, \hat{b}_{az}$ = estimates of the acceleration measurement biases.

These terms are numerically integrated by the equations

$$\dot{d\hat{x}}_i(t+\Delta t_f) = \dot{d\hat{x}}_i(t) + a_{si}\Delta t_f \quad ,$$

$$d\hat{x}_i(t+\Delta t_f) = d\hat{x}_i(t) + [\dot{d\hat{x}}_i(t+\Delta t_f)+\dot{d\hat{x}}_i(t)]\Delta t_f/2 \quad . \tag{A.23}$$

Here, the subscripts i refer to the three (x, y, and z) components of runway referenced, estimated change in position ($d\hat{x}_i$) and velocity ($\dot{d\hat{x}}_i$) due to acceleration (a_{si}). Equations (A.23) are approximations which are valid for a "flat" nonrotating earth. The errors resulting from this approximation are negligible in comparison to the errors caused by inertial hardware components (that is, the errors in the attitude and heading references and the errors in the body-mounted accelerometers). In the on board program, the raw acceleration data is accepted and integrated at 20 Hz (i.e., Δt_f is 0.05 sec).

The vector form of the error equations is given in Eq. (A.2) where the fourteen element error state vector, dx, is as defined previously. In the subsequent summary, it is assumed that elements of the noise vector $u(t_k)$ are all independent variables with unit variance. The actual magnitudes associated with the noise are included as constants of the Φ_u matrix of Eq. (A.4).

The transition matrix, Φ, is approximated as

$$\Phi = I + A \quad . \tag{A.24}$$

Here, I is the identity matrix, and A is a sparse matrix which represents the matrix F_x in discrete form. It is now defined for the x-y portions of the filter.

The x-y portion of the filter has ten elements which include:

dx(1) = position error component along runway (dx),

dx(2) = position error component normal to runway (dy),

dx(3) = velocity error component along runway ($v_x = \dot{dx}$),

dx(4) = velocity error component normal to runway ($v_y = \dot{dy}$),

dx(5) = acceleration bias component along runway (b_{ax}),

dx(6) = acceleration bias component normal to runway (b_{ay}),

dx(7) = TACAN range measurement bias (b_r),

dx(8) = TACAN bearing measurement bias (b_ψ),

dx(9) = wind error component along runway (w_x),

dx(10) = wind error component normal to runway (w_y).

In the subsequent discussion, for convenience, the details of the x-y portion of the filter are separated from the z portion by subscripts "x" and "z". For the horizontal x-y portion of the filter, the non-zero elements of A in Eq. (A.24) are given by

$$A_x(1,3) = A_x(2,4) = A_x(3,5) = A_x(4,6) = \Delta t \ ,$$

$$A_x(1,5) = A_x(2,6) = \Delta t^2/2 \ ,$$

$$A_x(5,5) = A_x(6,6) = \Delta t/\tau_a \ ,$$

$$A_x(7,7) = -\Delta t/\tau_r \ , \qquad \text{(A.25)}$$

$$A_x(8,8) = -\Delta t/\tau_\psi \ ,$$

$$A_x(9,9) = A_x(10,10) = -\Delta t/\tau_w \ .$$

Here,

Δt = period over which the transition matrix is used,

τ_a = time constant for acceleration colored noise (100 sec),

τ_r = time constant for TACAN range colored noise (1000 sec),

τ_ψ = time constant for TACAN bearing colored noise (1000 sec),

τ_w = time constant for wind error colored noise (100 sec).

Nominal values used for the time constants are shown in parentheses.

The nonzero elements of the forcing matrix Φ_{ux} (from Eq. (A.4)) are given as

$$\Phi_{ux}(3,3) = \Phi_{ux}(4,4) = \sigma_v \Delta t \ ,$$

$$\Phi_{ux}(5,5) = \Phi_{ux}(6,6) = \sigma_a \sqrt{2\Delta t/\tau_a} \ ,$$

$$\Phi_{ux}(7,7) = \Phi_r \sqrt{2\Delta t/\tau_r} \ , \qquad \text{(A.26)}$$

$$\Phi_{ux}(8,8) = \Phi_\psi \sqrt{2\Delta t/\tau_\psi} \ ,$$

$$\Phi_{ux}(9,9) = \Phi_{ux}(10,10) = \Phi_w \sqrt{2\Delta t/\tau_w} \ .$$

Here,

Δt = period of the major time update (1.5 sec),

σ_v = standard deviation (std) of velocity noise (0.0762 m/s),

σ_a = std of acceleration colored noise (0.1524 m/s^2),

σ_r = std of TACAN range colored noise (304.8 m),

σ_ψ = std of TACAN bearing colored noise (2 deg),

σ_w = std of wind colored noise (6.1 m/s).

The z filter has a four element error state comprised of

dx(1) = vertical position component error (dz),

dx(2) = vertical velocity component error ($v_z = \dot{dz}$),

dx(3) = vertical acceleration bias component error (b_{az}),

dx(4) = bias error in barometric altitude (b_h).

The non-zero elements of A_z in Eq. (A.24) for the z portion of the filter are as follows:

$$A_z(1,2) = \Delta t \quad ,$$

$$A_z(1,3) = \Delta t^2/2 \quad ,$$

$$A_z(2,3) = \Delta t \quad , \qquad \text{(A.27)}$$

$$A_z(3,3) = -\Delta t/\tau_a \quad ,$$

$$A_z(4,4) = -\Delta b/\tau_h \quad .$$

Here,

τ_h = time constant for barometric altimeter colored noise (1000 sec).

The non-zero elements of the forcing matrix (Φ_u of Eq. (A.3)) for the z-portion of the filter are given by

$$\Phi_{uz}(2,2) = \sigma_v \Delta t_k \quad ,$$

$$\Phi_{uz}(3,3) = \sigma_a \sqrt{2\Delta t/\tau_a} \quad , \qquad \text{(A.28)}$$

$$\Phi_{uz}(4,4) = \sigma_h \sqrt{2\Delta t/\tau_h} \quad .$$

Here,

σ_h = std of barometric altimeter bias colored noise (60.96 m).

Again, the nominal value for the standard deviation is given in parentheses.

External Measurement Processing Equations

To relate the external measurements to the estimated state $\hat{X}$, mathematical models of the measurements are required in terms of the elements of $\hat{X}$. The models are required for:

(1) defining the computed measurement as a function of the estimated state (i.e., $\hat{Y}(\hat{X},t_m)$ used in Eq. (A.15a),

(2) defining the partial row vector which relates the residual to the error state (i.e., H of Eqs. (A.8) and (A.16a)), and

(3) defining the variance of the random error in the measurement (i.e., Q of (A.17)).

The models used in the on board program are developed in this section for TACAN, MODILS, airspeed, and altitude measurements.

TACAN.- TACAN measurements consist of: (a) the range from the aircraft to the station, and (b) the bearing (with respect to magnetic north) of the station with respect to the aircraft. The range measurement is modeled as

$$Y_{tr} = \sqrt{(x-x_T)^2 + (y-y_T)^2 + (z-z_T)^2} + b_r + q_{tr} \quad . \quad (A.29)$$

Here,

x, y, z = the coordinates of the aircraft with respect to the runway reference frame,

x_T, y_T, z_T = the coordinates of the TACAN station with respect to the runway reference frame,

b_r = the bias error in the range measurement, and

q_{tr} = the random noise error in the range measurement.

The estimated measurement is computed from

$$\hat{Y}_{tr} = \sqrt{(\hat{x}-x_T)^2 + (\hat{y}-y_T)^2 + (\hat{z}-z_T)^2} + \hat{b}_r \quad , \qquad (A.30)$$

where $\hat{x}$, $\hat{y}$, and $\hat{b}_r$ are state variables obtained from the x-y filter, and $\hat{z}$ is obtained from the z filter.

The non-zero elements of the row vector H of Eq. (A.8) are calculated from

$$H_{trx}(1) = (\hat{x}-x_T)/(\hat{Y}_{tr}-\hat{b}_r) \quad ,$$

$$H_{trx}(2) = (\hat{y}-y_T)/(\hat{Y}_{tr}-\hat{b}_r) \quad , \qquad (A.31)$$

$$H_{trx}(7) = 1.$$

The variance Q_{tr} of the random noise error in the TACAN range measurement is assumed to be a constant given by

$$Q_{tr} = (92 \text{ m})^2 \quad . \qquad (A.32)$$

The bearing measurement is modeled as

$$Y_{tb} = \tan^{-1}\left[\frac{(y_T-y)}{(x_T-x)}\right] + \psi_r + b_\psi + q_{tb} \quad . \qquad (A.33)$$

Here,

ψ_r = the azimuth of the runway with respect to magnetic north,

b_ψ = the bias error in the bearing measurement, and

q_{tb} = the random noise error in the bearing measurement.

The estimated measurement is computed from

$$\hat{Y}_{tb} = \tan^{-1}\left[\frac{(y_T-\hat{y})}{(x_T-\hat{x})}\right] + \psi_r + \hat{b}_\psi \quad , \tag{A.34}$$

where $\hat{x}$, $\hat{y}$, and $\hat{b}_\psi$ are state variables of the x-y filter.

The non-zero elements of the row vector H for the bearing measurement are calculated from

$$H_{tbx}(1) = (y_T-\hat{y})/[(\hat{x}-x_T)^2 + (\hat{y}-y_T)^2] \quad ,$$

$$H_{tbx}(2) = (x-\hat{x}_T)/[(\hat{x}-x_T)^2 + (\hat{y}-y_T)^2] \quad , \tag{A.35}$$

$$H_{tbx}(8) = 1.$$

The variance Q_{tb} of the random noise error in the TACAN bearing measurement is assumed to be a constant given by

$$Q_{tb} = (1.\ \text{deg})^2 \quad . \tag{A.36}$$

MODILS range and azimuth.- The MODILS measurements used in the x-y portion of the Kalman filter are range and azimuth from a co-located DME transponder and azimuth scanner. The range measurement is modeled as

$$Y_{mr} = \sqrt{(x-x_m)^2 + (y-y_m)^2 + (z-z_m)^2} + q_{mr} \quad . \tag{A.37}$$

Here,

x_m, y_m, z_m = coordinates of the MODILS transponder and scanner with respect to the runway reference frame,

q_{mr} = the random noise error in the range measurement.

The estimated measurement is computed from

$$\hat{Y}_{mr} = \sqrt{(\hat{x}-x_m)^2 + (\hat{y}-y_m)^2 + (\hat{z}-z_m)^2} \quad . \tag{A.38}$$

Here, $\hat{x}$ and $\hat{y}$ are state variables obtained from the x-y filter, and $\hat{z}$ is obtained from the z filter.

The non-zero elements of the row vector H for the range measurement are calculated from

$$\begin{aligned} H_{mrx}(1) &= (\hat{x}-x_m)/\hat{Y}_{mr} \quad . \\ H_{mrx}(2) &= (\hat{y}-y_m)/\hat{Y}_{mr} \quad . \end{aligned} \tag{A.39}$$

The variance of the random noise error in the range measurement is assumed to be a constant given by

$$Q_{mr} = (18.3 \text{ m})^2 \quad . \tag{A.40}$$

The MODILS azimuth measurement is modeled as

$$Y_{ma} = \tan^{-1}[(y-y_m)/\sqrt{(x-x_m)^2 + (z-z_m)^2}\,] + q_{ma} . \tag{A.41}$$

Here, q_{ma} is a random error in the azimuth measurement.

The estimated measurement is computed from

$$\hat{Y}_{ma} = \tan^{-1}[(\hat{y}-y_m)/\sqrt{(\hat{x}-x_m)^2 + (\hat{z}-z_m)^2}\,] . \tag{A.42}$$

Again, $\hat{x}$ and $\hat{y}$ are state variables of the x-y filter, and $\hat{z}$ is obtained from the z filter.

The non-zero elements of the row vector H for the azimuth measurement are given by

$$\begin{aligned} H_{max}(1) &= (\hat{y}-y_m)(\hat{x}-x_m)/(r_1(r)^2) \quad , \\ H_{max}(2) &= r_1/r^2 \quad . \end{aligned} \tag{A.43}$$

Here, r and r_1 are defined as

$$r = \sqrt{(\hat{x}-x_m)^2 + (\hat{y}-y_m)^2 + (\hat{z}-z_m)^2} \quad ,$$

$$r_1 = \sqrt{(\hat{x}-x_m)^2 + (\hat{z}-z_m)^2} \quad . \tag{A.44}$$

The variance of the random error in the measurement is assumed to be a constant given by

$$Q_{ma} = (0.1 \text{ deg})^2 \quad . \tag{A.45}$$

True airspeed.- The existing STOLAND software computes the level components of true airspeed in the runway reference frame from air data and altitude data. These components are assumed to be direct measurements in the Kalman filter's x-y frame rather than using the more complex mechanization involving actual raw data sensors.

The x and y component air data measurements of true airspeed are modeled as

$$Y_{ax} = v_x - w_x + q_{ax} \quad ,$$

$$Y_{ay} = v_y - w_y + q_{ay} \quad . \tag{A.46}$$

Here,

v_x, v_y = ground velocity components along and normal to the runway,

w_x, w_y = wind velocity components along and normal to the runway,

q_{ax}, q_{ay} = the random noise errors in the air data measurements.

The estimated measurements are computed from

$$\hat{Y}_{ax} = \hat{v}_x - \hat{w}_x \quad ,$$

$$\hat{Y}_{ay} = \hat{v}_y - \hat{w}_y \quad , \tag{A.47}$$

where $\hat{v}_x$, $\hat{v}_y$, $\hat{w}_x$, and $\hat{w}_y$ are state variables of the filter.

The non-zero elements of the row vector H for along the runway are given by

$$H_{axx}(3) = 1 \ ,$$

$$H_{axx}(9) = -1. \qquad \text{(A.48a)}$$

The non-zero elements of the row vector H for normal to the runway are given by

$$H_{ayx}(4) = 1.$$

$$H_{ayx}(10) = -1. \qquad \text{(A.48b)}$$

The variances of the random noise error in the air data measurements are assumed to be constants given by

$$Q_{ax} = Q_{ay} = (0.61 \text{ m/s})^2 \ . \qquad \text{(A.49)}$$

Barometric altimeter.- The barometric altimeter measurement is modeled as

$$Y_h = -z + h_r + b_n + q_h \ . \qquad \text{(A.50)}$$

Here,

z = vertical position of the aircraft with respect to the runway reference,

h_r = runway altitude with respect to sea level,

b_h = bias error in the barometric altitude measurement,

q_h = random noise error in the barometric altitude measurement.

The estimated measurement is computed from

$$\hat{Y}_h = -\hat{z} + h_r + \hat{b}_h \ , \qquad \text{(A.51)}$$

where $\hat{z}$ and $\hat{b}_h$ are state variables of the z portion of the filter. The non-zero elements of the row vector H are given by

$$H_{hz}(1) = -1 \quad ,$$
$$H_{hz}(4) = 1. \qquad (A.52)$$

The variance of the random noise error in the measurement is assumed to be a constant given by

$$Q_h = (1.2 \text{ m})^2 \quad . \qquad (A.53)$$

Radio altimeter.- The radio altimeter measurement is modeled as

$$Y_r = -z + q_r \quad , \qquad (A.54)$$

where q_r is random noise error in the radio altimeter measurement. The estimated measurement is computed from

$$\hat{Y}_r = -\hat{z} \quad . \qquad (A.55)$$

The non-zero element of the row vector H is

$$H_{rz}(1) = -1 \quad . \qquad (A.56)$$

The variance of the random noise error in the measurement is assumed to be a constant given by

$$Q_r = (0.6 \text{ m})^2 \quad . \qquad (A.57)$$

MODILS elevation.- Define the aircraft relative position coordinates with respect to the MODILS elevation antenna as

$$x_e = x - x_E \quad ,$$

$$y_e = y - y_E \quad , \qquad \text{(A.58)}$$

$$z_e = z - z_E \quad .$$

Here, (x_E, y_E, z_E) are the location components of the MODILS elevation antenna with respect to the runway reference frame. Also, define the auxiliary quantities

$$z_1 = z_e \cos(5°) - x_e \sin(5°) \quad ,$$

$$x_1 = x_e \cos(5°) + z_e \sin(5°) \quad , \qquad \text{(A.59)}$$

$$r_1 = \sqrt{x_1^2 + y_e^2} \quad .$$

The altitude measurement calculated from the MODILS elevation measurement is expressed as

$$Y_e = [-x_e \sin(5°) + r_1 \tan(\varepsilon - 5°)]/\cos(5°) + q_e \quad . \qquad \text{(A.60)}$$

In Eq. (A.60),

ε = the elevation measurement above the horizontal plane,

q_e = the random noise error in the pseudo-altitude measurement.

Equation (A.60) requires the values of x_e, y_e, and z_e, which are not available; instead, estimates of these states are used in the calculation.

The estimated measurement is given by

$$\hat{Y}_e = -\hat{z} \quad . \qquad \text{(A.61)}$$

The non-zero element of the row vector H is given by

$$H_{ez}(1) = -1 \ . \tag{A.62}$$

The variance of the random noise error in the measurement is assumed to be a range dependent quantity

$$Q_e = (0.002\ r_1)^2 \ , \tag{A.63}$$

where r_1 of Eq. (A.59) is in meters.

Measurement Preprocessing and Rejection

The mechanized filter contains routines for calculating the residuals and partials (H vector) as just discussed, and for summing the results appropriately at a 10 Hz frequency. Each residual sum and its partial are transferred to appropriate arrays for processing by the Kalman filter algorithm at the basic 0.667 Hz frequency.

The preprocessing routines contain logic for executing the following steps in a sequential manner for each measurement:

(1) Test the hardware validity flags. If the measurement is invalid, the subsequent steps are bypassed. This step is omitted for the airspeed and barometric altitude measurements because they do not have hardware validity flags.

(2) Compute the residual by $\Delta y_i = Y_i - \hat{Y}_i$ (measurement minus computed measurement).

(3) Test the reasonableness of the residual. If the residual magnitude exceeds a precomputed tolerance level, the subsequent steps are bypassed.

(4) Accumulate the residual into the residual sum by $y_{si} = y_{si} + \Delta y_i$.

(5) Calculate the H vector for the i^{th} measurement, and reference the vector to time i_{t_k} by $H_{mi}(t_k) = H_i(t)\ \Phi(t;t_k)$.

(6) Accumulate elements of H_{mi} into the partial sum by $H_{msi} = H_{msi} + H_{mi}$.

(7) Increment a measurement counter by unity. (The number of valid measurements in each sum is calculated.)

The TACAN bearing and MODILS azimuth measurements have additional logic before Step (2) which rejects the measurements if the ground distance from the station (or scanner) to the aircraft is less than 305 meters.

Following completion of the above logic for each of the measurements (every 0.1 sec), a marker is tested to determine if the basic 1.5 sec basic cycle is complete. If this test is passed, the incremental state changes are calculated, as is described earlier.

The variance of the random error in each residual sum is calculated from

$$Q^i = Q^i_{xx}(n^i_x)^{1.4} \qquad \text{(A.64)}$$

where

Q^i_{xx} = the variance for an individual measurement,

n^i_x = the number of residuals in the sum.

The number 1.4 is used to account for the fact that the random error q in each measurement has some correlation from time point to time point.

In addition to the validity flags and residual reasonableness tests, a test is made on the reasonableness of the residual sum before it is used to calculate an incremental state change. The Potter algorithm (see Eq. (A.14)) requires calculation of the quantity

$$(\sigma_m)^2 = B^T B + Q \qquad \text{(A.65)}$$

for each residual sum. Let

y_{sm} = the residual sum for the particular standard deviation σ_m involved.

Then the mechanized filter rejects the measurement if

$$c|y_{sm}| > \sigma_m \quad . \tag{A.66}$$

The value of c used in this study was 0.25. The value of σ_m for each measurement is used in the reasonableness test prior to summing the residual (Step (3) above).

Interface With STOLAND

The estimated error state is obtained from the x-y and z filters every 1.5 sec. When the error state is added to the state estimate, discrete jumps occur which, as a result of the low frequency, may be objectionable to the pilot or automatic control system. In order to prevent discrete changes from occurring in the state estimate used by the STOLAND system, smoothing logic was defined as depicted in Fig. 10. This logic is explained here. Let

$\hat{x}_r$ = estimated position vector,

$\hat{v}_r$ = estimated velocity vector,

c_x = position smoothing vector,

c_v = velocity smoothing vector.

Assume that errors $d\hat{x}_r$ and $d\hat{v}_r$ have been estimated (from the filters) to be added to $\hat{x}_r$ and $\hat{v}_r$, respectively. Then at the time of introduction (time point t_{k+1}), the following equations are executed:

$$\begin{aligned}
(\hat{x}_r)_a &= (\hat{x}_r)_b + d\hat{x}_r \quad , \\
(\hat{v}_r)_a &= (\hat{v}_r)_b + d\hat{v}_r \quad , \\
(c_x)_a &= (c_x)_b - d\hat{x}_r \quad , \\
(c_v)_a &= (c_v)_b - d\hat{v}_r \quad .
\end{aligned} \tag{A.67}$$

Then, at a 20 Hz frequency, c_x and c_v are decremented in accordance with

$$c_x(t+\Delta t_f) = \alpha c_x(t) \quad ,$$
$$c_v(t+\Delta t_f) = \beta c_v(t) \quad . \tag{A.68}$$

Here, α and β are constants computed from

$$\alpha = e^{-.05/\tau_x} \quad ,$$
$$\beta = e^{-.05/\tau_v} \quad , \tag{A.69}$$

and (τ_x, τ_v) are time constants for decaying the c_x and c_v quantities.

Now define the smoothed position and velocity vectors as

$$\hat{x}_s = \hat{x}_r + c_x \quad ,$$
$$\hat{v}_s = \hat{v}_r + c_v \quad . \tag{A.70}$$

As may be seen by inserting Eq. (A.67) into Eq. (A.70),

$$(\hat{x}_s)_a = (\hat{x}_s)_b \quad ,$$
$$(\hat{v}_s)_a = (\hat{v}_s)_b \quad . \tag{A.71}$$

Thus, the discrete change $(d\hat{x}_r, d\hat{v}_r)$ does not cause a jump in the smoothed state estimate $(\hat{x}_s, \hat{v}_s)$ used by the pilot and automatic control system. Also, the error state is added in a smooth manner using Eqs. (A.67) and (A.70) as the c_x and c_v smoothing vectors decay. The values for the time constants τ_x and τ_v were both selected to be 5 sec.

Filter Initialization

The filter mechanization was arranged such that the start of initialization or reinitialization occurs consistent with that of the existing STOLAND complementary filter. This was done to give valid comparison of the performance of the two filters during the simulation and flight test phases.

The initialization of the Kalman filter consists of the following:

(1) setting the position (x,y,z) state variables from TACAN and barometric altimeter data,

(2) setting the velocity (v_x,v_y,v_z) state variables at the runway referenced true airspeed values ($v_z = 0$),

(3) setting wind, acceleration bias, TACAN measurement bias and baro-altimeter state variables zero, and

(4) setting the initial square root matrix in a manner consistent with the above.

The position components are calculated from

$$x = x_t - r_c \cos(\Delta\psi) \quad , \qquad (A.72)$$

$$y = y_t - r_c \sin(\Delta\psi) \quad .$$

Here,

$$r_c = \sqrt{(Y_{tr})^2 - (h_t)^2} \quad ,$$

h_t = altitude above the TACAN station computed from barometric altitude,

$$\Delta\psi = Y_{tb} - \psi_r \quad .$$

The velocity components are given by,

$$v_x = Y_{ax} \quad ,$$

$$v_y = Y_{ay} \quad , \qquad (A.73)$$

$$v_z = 0 \quad .$$

The non-zero elements of the initial square root covariance matrix W_x for the x-y filter are given by the following:

$W_x(1,1) = \cos(\Delta\psi)\sigma_r$, σ_r = std of TACAN range bias $\cong$ 305 m.

$W_x(1,2) = \sin(\Delta\psi)\sigma_r$,

$W_x(1,7) = \sigma_r$,

$W_x(2,1) = \cos(\Delta\psi)\sigma_{qr}$, σ_{qr} = std of random noise error in the TACAN range measurement $\cong$ 37m.

$W_x(2,2) = \sin(\Delta\psi)\sigma_{qr}$,

$W_x(3,1) = (y_T-y)\sigma_\psi$, σ_ψ = std of TACAN bearing bias $\cong$ 2 deg.

$W_x(3,2) = (x_T-x)\sigma_\psi$,

$W_x(3,8) = \sigma_\psi$,

$W_x(4,1) = (y_T-y)\sigma_{q\psi}$, $\sigma_{q\psi}$ = std of random noise error in TACAN bearing.

$W_x(4,2) = (x_T-x)\sigma_{q\psi}$,

$W_x(5,3) = -\cos(\psi_i-\psi_r)\sigma_{va}$, σ_{va} = std of random error in airspeed measurement $\cong$ 0.61 m/sec.

$W_x(5,4) = -\sin(\psi_i-\psi_r)\sigma_{va}$,

$W_x(6,3) = v_y\sigma_{\psi i}$, $\sigma_{\psi i}$ = std of bias error in initial heading $\cong$ 2 deg.

$W_x(6,4) = -v_x\sigma_{\psi i}$,

$W_x(7,3) = \sigma_{wx}$, σ_{wx} = std of x component of wind $\cong$ 6.1 m/sec.

$W_x(7,9) = \sigma_{wx}$,

$W_x(8,4) = \sigma_{wy}$, σ_{wy} = std of y component of wind $\cong$ 6.1 m/sec.

$W_x(8,10) = \sigma_{wy}$,

$W_x(9,5) = \sigma_{ax}$,

$W_x(10,6) = \sigma_{ay}$,

$\sigma_{ax,y}$ = acceleration random error std = .3 m/sec^2.

The term ψ_i above is the magnetic heading measurement at the time of initialization.

The non-zero elements of the initial square root covariance matrix W_z for the z filter are as follows:

$W_z(1,1) = \sigma_{hb}$,

$W_z(1,4) = \sigma_{hb}$,

σ_{hb} = std for bias error in baro altitude $\cong$ 61 m.

$W_z(2,2) = \sigma_{va}$,

σ_{va} = std of initial vertical error $\cong$ (.3 m/sec).

$W_z(3,3) = \sigma_{az}$,

σ_{az} = vertical acceleration random noise error std $\cong$ .3 m/sec^2.

$W_z(4,1) = \sigma_{hr}$,

σ_{hr} = std of random error in baro-altimeter $\cong$ 1.2 m.

APPENDIX B

DESCRIPTION OF THE AIRBORNE NAVIGATION SYSTEM MECHANIZATION

The navigation system employing the Kalman filter described in Appendix A was designed to operate as an experiment on the Twin Otter aircraft by using the leftover memory and real time of the STOLAND avionics system software. The experimental objectives were to compute and record the navigation outputs from the Kalman filter by using the same raw data sources as were used by the STOLAND complementary filter. The complementary filter outputs were also computed and recorded at the same time (in a time-sharing sense) as those of the Kalman filter; this allowed a direct comparison of the two filters' performances.

This appendix describes the mechanized Kalman filter algorithm logic of the airborne navigation program and the executive logic which provided the time-sharing between the original STOLAND software and the Kalman filter.

Executive Driver

In order to provide the available real time to mechanize the Kalman filter logic, it was necessary to develop a new executive for the STOLAND software used on the Twin Otter aircraft. A macro flowchart of this executive is presented in Fig. B.1. At the beginning of program operation, initialization logic is executed. The interrupts are then enabled, and the Far Background logic is entered.

In this system, the Far Background logic is used to execute lowest priority computations and to mark time until various interrupt signals indicate that it is time for faster synchronous computations to begin. The program waits in the Far Background routine until an interrupt occurs. The STOLAND system 1819A computer software contains several levels and sources of program interrupt. However, only the main stream of calculations, triggered by the 1 kHz internal clock and pertinent to the Kalman filter mechanization are described here. A clock interrupt causes transfer of program operation to a location where a counter is decremented and tested to determine if it is time to initiate the 20 Hz calculations. If not, the program returns to the location where interrupted, and it then continues execution.

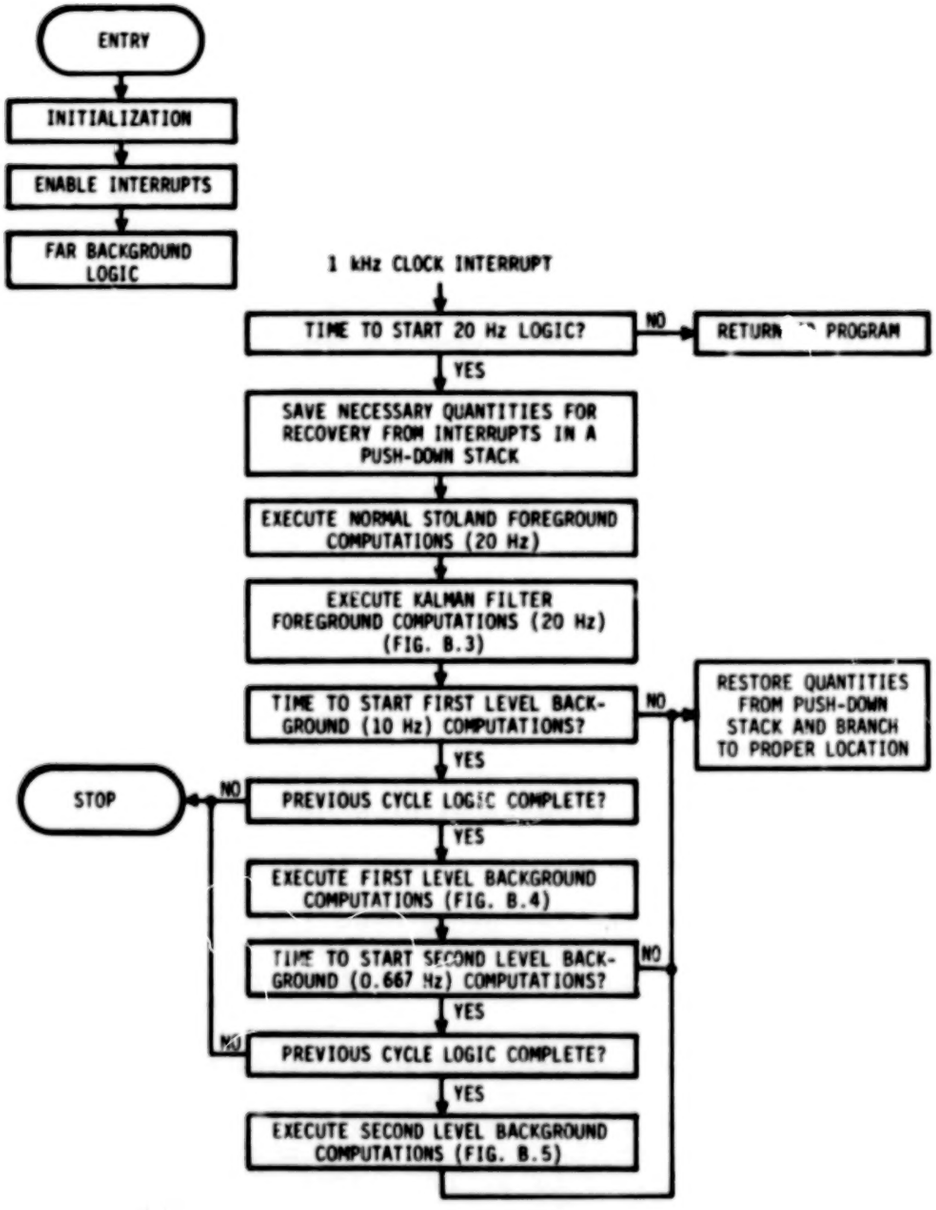


FIGURE B.1.- MACRO FLOWCHART OF NEW EXECUTIVE FOR STOLAND SOFTWARE TO INCLUDE THE KALMAN FILTER EQUATIONS.

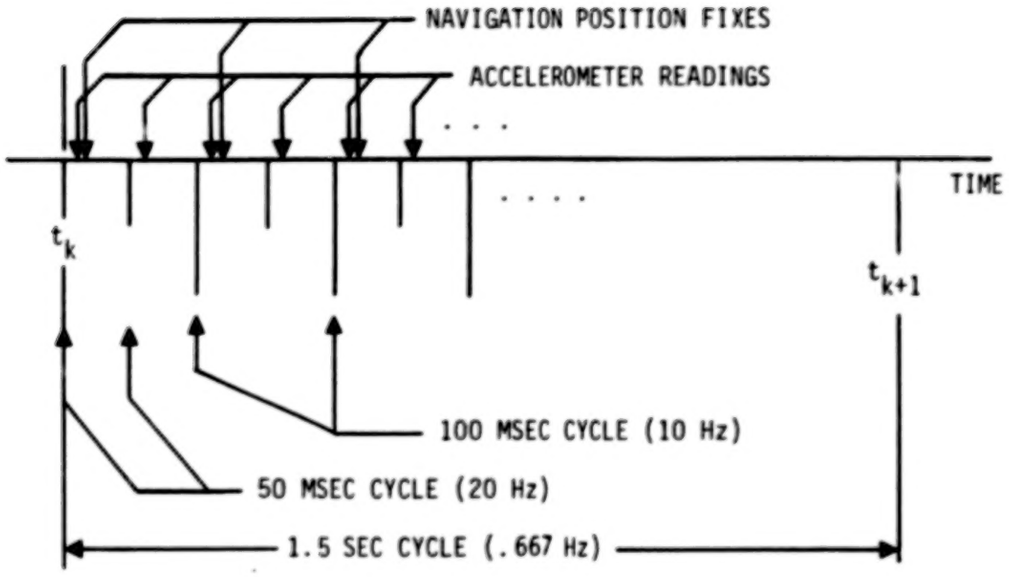
If it is time to start the 20 Hz computations, then the necessary quantities required for recovery from the interrupt (register contents, program location at interrupt) are saved in a software push-down stack. The program then executes the normal STOLAND (foreground) Twin Otter navigation, guidance, control and display computations. The STOLAND avionics equations are sectioned such that there always remains a small essentially unused period of time (> 1 msec) at the end of each 50 msec computation cycle. This period was used to mechanize the Kalman filter. Following completion of the STOLAND Foreground equations, the program executes the logic for interfacing the Kalman filter logic and algorithms with the STOLAND algorithms.

The Kalman filter equations are divided into three priority levels executed at rates of 20 Hz, 10 Hz and 0.667 Hz. These priority levels are summarized in equation form in Table 2 of Chapter III. They are referred to as Foreground, First Level Background, and Second Level Background equations. These priority levels are used to allow integrating the accelerometer readings at a high rate (10 Hz), and accumulating the position and air data measurements (fixes) and preprocessing them at a lower rate (10 Hz). There was not real time available to execute the entire Kalman filter at this speed. Thus, the majority of the Kalman filter computations are spread over the real time available during a 1.5 sec period (0.667 Hz).

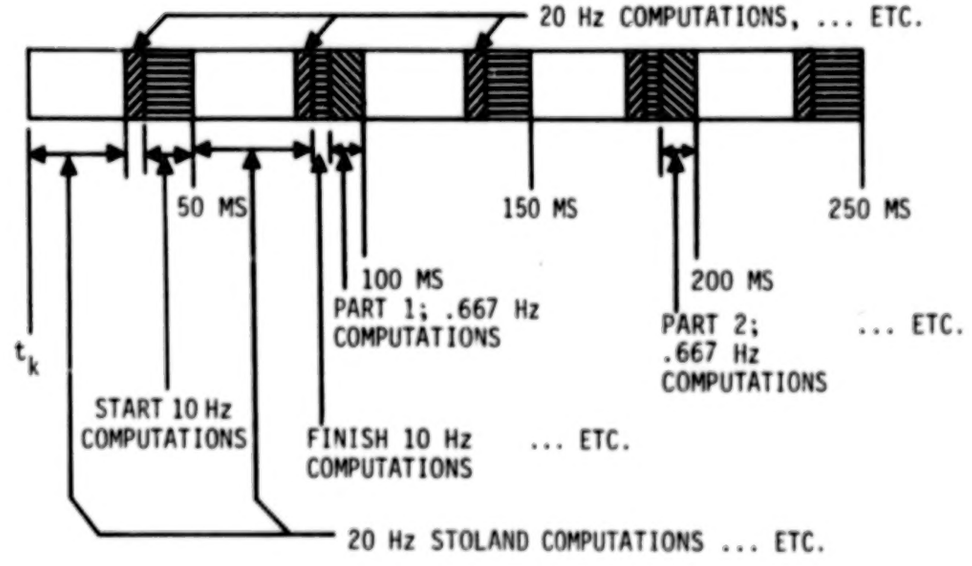
Figure B.2 illustrates the cycles used in mechanizing the Kalman filter. Also illustrated is the division of time spent in each level of computation of five example consecutive 50 msec computation cycles. The logic which interconnects each priority level in the Kalman filter computations is explained further in the following discussion.

Following completion of the Foreground computations, the next logic step determines if it is time to start the First Level Background (10 Hz) calculations every other cycle of the Kalman filter. If not, the executive restores all the quantities saved in the push-down stack and branches to the saved interrupt location. In this instance, the program branches to the higher priority computations in the order: (1) 10 Hz (if not complete); (2) 0.667 Hz (if not complete); or (3) Far Background.

If the 10 Hz computations are to be initiated, a marker is tested to see if the previous 10 Hz cycle calculations



(a) Mechanization Cycles



(b) Example (Distorted, hypothetical) Time Sharing in Five Consecutive 50 msec Computation Cycles

FIGURE B.2.- ILLUSTRATION OF MECHANIZATION CYCLES AND SEQUENTIAL PERIODS SPENT IN EACH COMPUTATION LEVEL.

were completed before the interrupt. If not, a STOP* is executed because either a software or hardware problem has prevented completion of the 10 Hz logic within the allowed time. With no malfunction, the program proceeds to execute the 10 Hz Kalman filter computations shown in Table 2.

Following completion of the 10 Hz computations, the next logic step determines if the Second Level Background (0.667 Hz) Kalman filter computations shown in Table 2 should be initiated. If not, the executive again restores all the quantities saved in the push-down stack and branches to the saved interrupt location. In this instance, the program branches to the higher priority computations in the order: (1) 0.667 Hz (if not complete); (2) Far Background.

If the 0.667 Hz computations (Second Level Background) should be initiated, a marker is tested to see whether the previous 0.667 Hz cycle computations were completed in the allowed time. If not, a STOP is executed indicating that a hardware or software problem exists. If no malfunction exists, the 0.667 Hz computations are initiated.

Following completion of the 0.667 Hz computations, the executive restores the quantities from the push-down stack and branches to the saved location in the Far Background computations.

Foreground (10 Hz) Logic and Computations

The Kalman filter logic and computations which are executed at 20 Hz, are shown in Fig. B.3. The first logic test determines if artificial data should be generated internally for a straight-line path. This logic was used for check-out phases, and it is a convenient way of checking the Kalman filter operations for new assemblies.

For normal operation, the logic first transfers the raw acceleration data from the STOLAND complementary filter locations to the Kalman filter storage locations at a 20 Hz rate. Then, a logic test determines if the particular entry is at the starting time for a 10 Hz cycle. If true, the navigation aid measurements and their validity flags are transferred

*The STOP was only used in the ground tests of the system in the STOLAND laboratory. In the airborne mechanization, the STOP was replaced with a branch to the RESTORE QUANTITIES FROM PUSH DOWN STACK block shown on Fig. B.1. The airborne mechanization tries to complete the logic by skipping an interval.

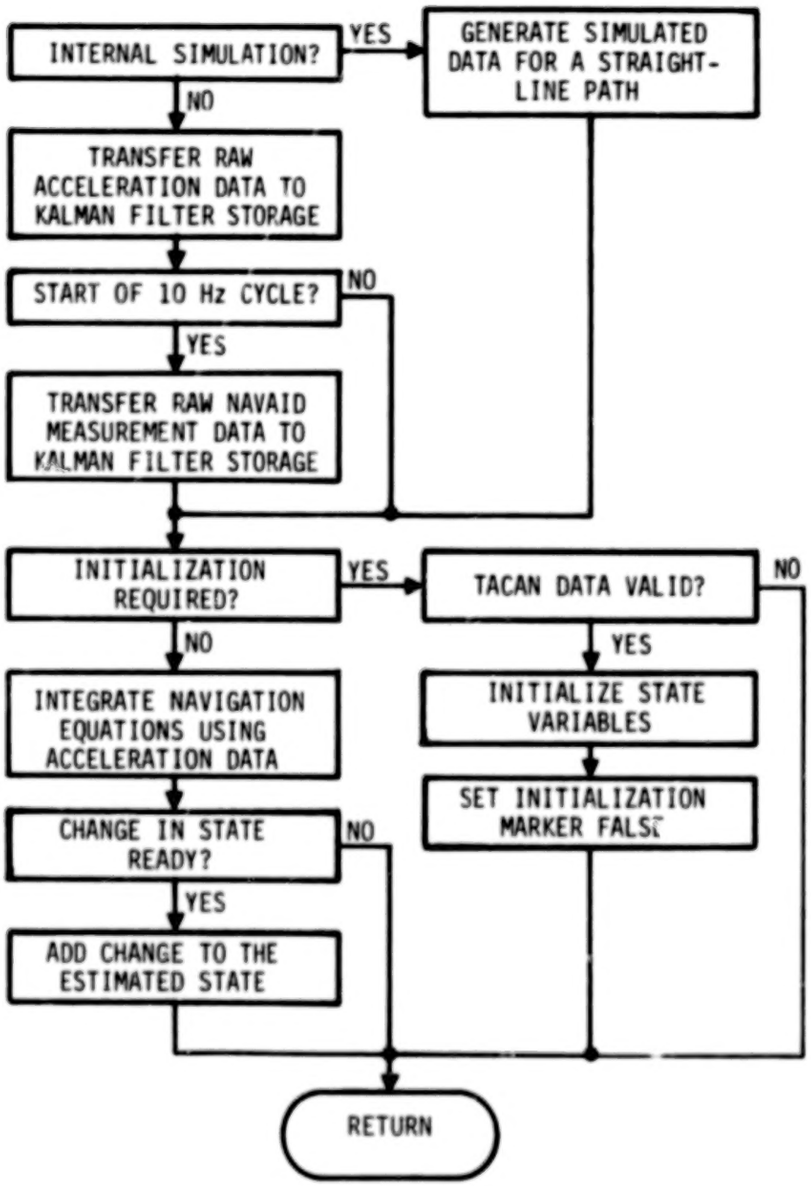


FIGURE B.3.- MACRO FLOW CHART OF KALMAN FILTER FOREGROUND LOGIC

from STOLAND locations used by the complementary filter to locations used by the Kalman filter.

Next, the navigation equations are executed. The first operation determines if initialization is required. This marker is set true when the regular STOLAND complementary filter begins its initialization. If initialization is required, then the validity flags for the TACAN data are tested. If both the range and bearing are valid, the state variables are initialized, and the initialization marker is set false. If TACAN data are not valid, the routine returns to the main executive program.

If initialization is not required, the position and velocity equations are integrated using the acceleration data. Following the integration, a marker is tested to determine if a state change (update) is ready. If it is ready, then the incremental change is added to the estimated state. Then, the program control returns to the main program.

First Level Background (10 Hz) Logic and Computations

Figure B.4 shows the navigation aid measurement preprocessing logic which is executed at 10 Hz. A marker is tested first to see if the preprocessing logic has been initialized. This marker is set true after the state variables have been initialized in the Foreground navigation equations (Fig. B.3). After initialization (if it is required), the logic does the necessary preprocessing for the TACAN, MODILS, air data, and radar altimeter measurements as described in Appendix A. The transition matrix elements are then updated so the matrix is valid for the next entry.

The incremental state change marker is then tested. If true, the residual sums are corrected, and the incremental state change is updated with the transition matrix. The change ready marker is then set true for use by the Foreground logic.

Next, the residual sum cycle marker is tested. Every fifteenth entry (1.5 sec), the marker is set true. Then, the residual sums and partials are transferred for use by the Second Level Background equations. Following transfer, the residual and partial sum locations are cleared, and program control returns to the main program.

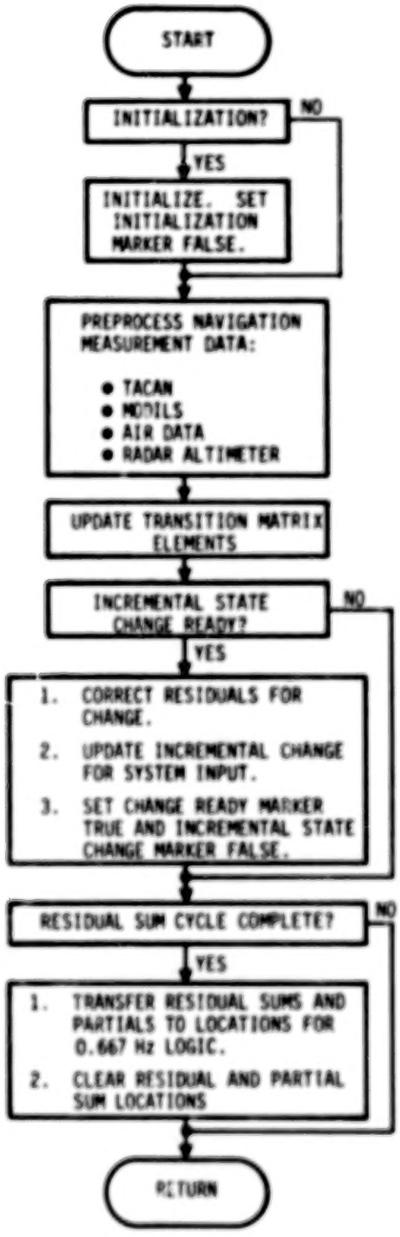


FIGURE B.4.- MACRO FLOWCHART OF FIRST LEVEL BACKGROUND LOGIC.

Second Level Background (0.667 Hz) Logic and Computations

Figure B.5 shows the Kalman filter logic which is executed at 0.667 Hz. The first test determines if initialization is required. If true, the square root covariance, transition matrix, and forcing function matrix are initialized, and the initialize marker is set false before return.

If the initialize marker is false, a test is made to determine if there are any measurements (residual sums) to be processed. If true, Potter's algorithm (see Appendix A) is used to process each available residual sum in a sequential manner. After completion, the incremental state change marker is set true for use in the First Level Background logic.

Finally, the square root matrix is updated to the beginning of the next measurement accumulation period (see Appendix A). Then, program control is again returned to the main executive.

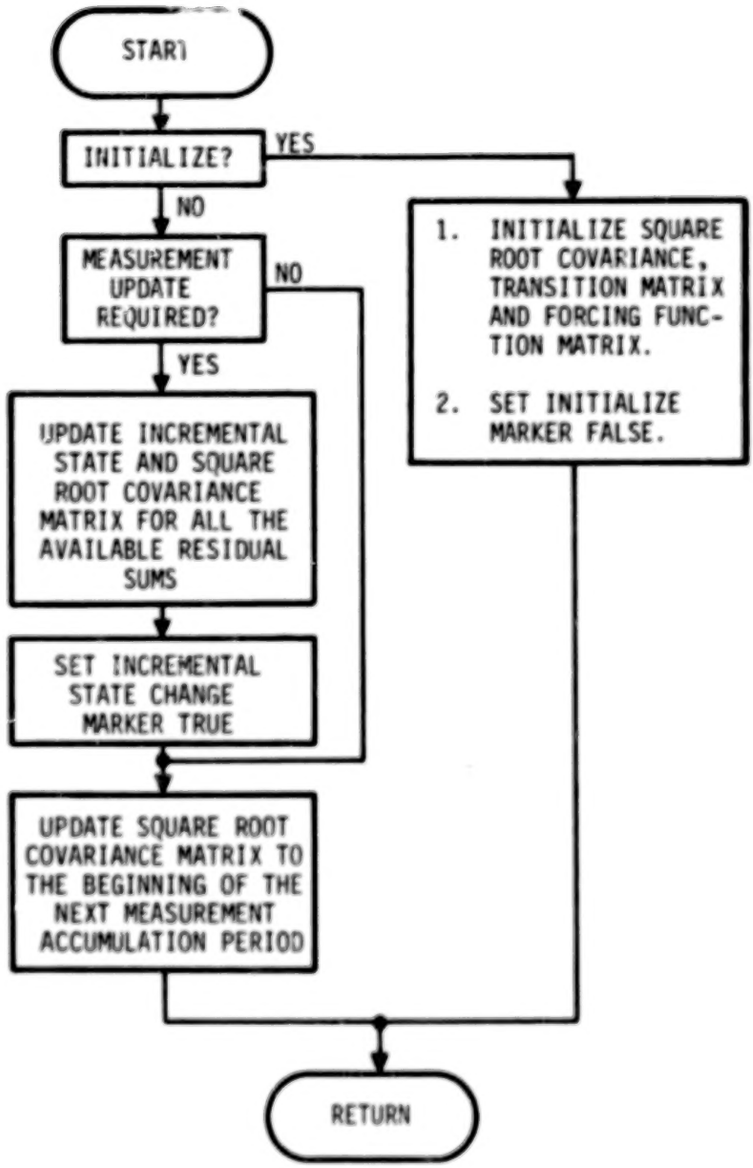


FIGURE B.5.- MACRO FLOWCHART OF THE SECOND LEVEL BACKGROUND LOGIC

REFERENCES

1. Neuman, F.; and Warner, D.N. Jr: A STOL Terminal Area Navigation System. NASA TMX-62,348, May 1974.

2. Desirazu, R.: Augmentor Wing and Twin Otter Navigation Program Document. Report No. 5440-0220-P110, Sperry Flight Systems, Phoenix, Arizona, November 1975.

3. Schmidt, S.F.: Precision Navigation for Approach and Landing Operations, 1972 JACC, Stanford University, Stanford, California, August 1972.

4. McGee, L.A.; et al: Flight Results From a Study of Aided Inertial Navigation Applied to Landing Operations. NASA TN D-7302, 1973.

5. Schmidt, S.F.; A Kalman Filter for the STOLAND System. NASA CR-137668, April 1975.

6. Schmidt, S.F.; and Mann, F.I.: A Three-Axis Kalman Filter for the STOLAND Flight Test System. NASA CR-137939, May 1976.

7. Kanning, G., Cicolani, L.S.; and Schmidt, S.F.: Application of Kalman Filtering for Terminal Area Navigation. NASA TMX (to be published).

8. Smith, D.W.; Neuman, F.; Watson, D.M.; and Hardy, G.H.: A Flight Investigation of a Terminal Area Navigation and Guidance Concept for STOL Aircraft. NASA TMX-62,376, July 1974.

9. Grgurich, J.; and Bradbury, P.: STOLAND Final Report. NASA CR-137972, 1976.

10. Kaminski, P.G.; Bryson, A.E., Jr.; and Schmidt, S.F.: Discrete Square Root Filtering - A Survey of Current Techniques. IEEE Transactions on Automatic Control. December 1971.

11. Bryson, A.E.; and Ho, Y.C.: Applied Optimal Control. Blaisdell, Waltham, Massachusetts, 1969.

12. Higgins, W.T.: A Comparison of Complementary and Kalman Filtering. IEEE Transactions on Aerospace and Electronic Systems. May 1975, pp. 321-325.

13. Goka, T.: A Study of Complementary Filters for STOLAND. AMA Report 77-25, Analytical Mechanics Associates, Mountain View, California, December 1977.

14. Schmidt, S.F.; Bjorkman, W.S.; and Conrad, B.: New Mechanization Equations for Aided Inertial Navigation Systems. NASA CR-2352. December 1973.

END